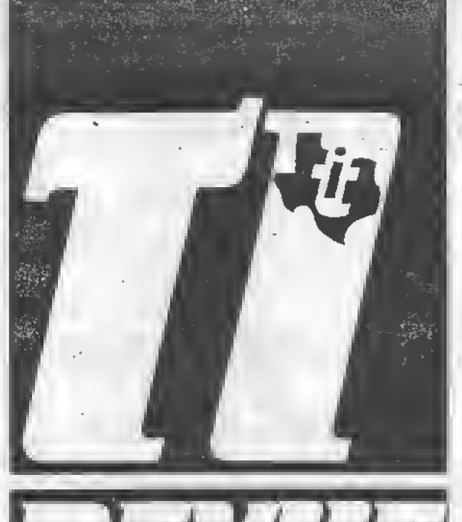


Nr. 9/85 — 4. Quartal — DM 5,50/ÖS 46/SFR 5,50

32 Kilobyte für 75 DM!

Das Magazin für TI 99-4A

30 Seiten Listings für Ihren TI 99/4A

Assembler-Kurs

So steuern Sie jeden Drucker richtig an

Speicher-Geheimnisse des TI

List-Dateien nicht mehr abtippen

IMPRESSUM

TI-REVUE, die Zeitschrift für den TI PC und den TI 99/4A erscheint monatlich im TI/CBM Verlag Werner E. Seibt, Postfach 1107, Elisabethstraße 1, 8044 Lohhof.
Redaktion: Senator-Presse-Service.
Verantwortlich für den Inhalt: Klaus Herdin und Heiner Martin.
Verantwortlich für Anzeigen: Bruno Redase.
Verantwortlich für Listings: Hartmut König.
Alle: Postfach 1107, 8044 Lohhof.
Anfragen bitte nur schriftlich.
Druck: Maier und Söhne
Es gilt die Honorarliste des Verlages.
Für unaufgefordert eingesandte Manuskripte und Listings keine Haftung.
Bei Einsendung von Texten, Fotos und Programmträgern erteilt der Autor dem Verlag die Genehmigung für einen einmaligen Abdruck sowie die Aufnahme in den Programm-Service nach den Verlags-Sätzen!

Kein Anspruch auf Lieferung bei Ausfall durch höhere Gewalt.
Gerichtsstand: München
Geschäftsführer: Werner E. Seibt
Abo- und Kassetten-Service: Henny Rose Seibt

Die nächste TI-REVUE am 27. Dez.

Wie immer steht unseren Lesern unser Telefon-Service zur Verfügung! Jeden Dienstag von 15 bis 19 Uhr. Für technische Fragen: 0731/33220 und für Listings/Programme: 089/1298013

INHALT

Nr. 9/85
Dezember

DIALOG

SERIE & SERVICE

TIPS & TRICKS

LISTINGS

Grüß Gott–Gruezi–Guten Tag

Die Umstellung auf monatliche Erscheinungsweise kostet uns natürlich doch einige Nerven und zusätzliche Arbeit. Immerhin muß ja jetzt von der Redaktion und im Verlag in gleicher Zeit die doppelte Seitenzahl bereitgestellt werden. Dennoch hoffen wir, daß auch diese Ausgabe Ihrer TI-REVUE wieder bei Ihnen Anklang findet.
Sehr erfreut ist momentan wohl die Lage auf dem Zubehörsektor für den TI 99/4A zu beurteilen. In den letzten Monaten wurden gerade aus deutscher Fertigung einige interessante Neuheiten angekündigt und auch teilweise schon ausgeliefert. Um so bemerkenswerter ist dies, da ja doch schon ein klein bißchen Unsicherheit vorhanden war, ob überhaupt und wenn ja zu welchem Preis in Zukunft Erweiterungen zu kaufen sind. Nun, da auch die Preise unter denen der Originalgeräte liegen,ist damit für die Zukunft vorgesorgt. Einen kleinen Mangel gibt es wohl an neuer Software, wie uns auch schon Händler ihr Leid geklagt haben. Aus neueren amerikanischen Katalogen können wir aber ersehen, daß jenseits des großen Teiches gerade hier viel auf den Markt gekommen ist, so daß sich die Situation hierzulande wohl bald bessern wird. In den USA scheint man momentan bei den Hardwareerweiterungen etwas hinterherzuhinken.
Größeren Wirbel, sprich Leseranfragen, hat die Veröffentlichung einer anderen Zeitschrift über einen neuen TI-Computer verursacht. Unsere Antwort dazu lesen Sie in der Rubrik „Leserfragen".
Nachzutragen dazu bleibt noch, daß dieser von der amerikanischen Firma MYARC entwickelte Computer auf einer Ausstellung in Chicago am 2. November gezeigt werden soll (der Redaktionsschluß für diese Ausgabe lag schon vor diesem Termin), in einer Version mit 256 KByte RAM. Dennoch, solange dafür kein Preis festgelegt und die Lieferfähigkeit vorhanden ist, handelt es sich eben um ein schönes Einzelstück, wie ja auch die amerikanische Firma CorComp schon einmal ein sehr teures Vorführmodell gezeigt hat, welches bisher nicht in Produktion ist. Von der Ausstellung in Chicago, die sich übrigens nur um den TI 99/4A dreht, haben wir einen Bericht von unserem amerikanischen Korrespondenten angefordert und hoffen, diesen in der nächsten Ausgabe veröffentlichen zu können. Bis dahin die besten Grüße von Eurem TI-REVUE-Team

Nicht vergessen: Seid Ihr mit dem Heft zufrieden, sagt es weiter, seid Ihr unzufrieden, sagt es uns.
Und: Jeden Dienstag von 15 bis 19 Uhr stehen Ihnen unser Lesertelefon zur Verfügung: Für technische und Assemblerfragen Tel.-Nr. 0731/33220 und zu den Listings Tel.-Nr. 089/1298013. Wir bitten um Verständnis, daß am 24. und 31.12.85 sowie am 6.1.86 das Lesertelefon nicht besetzt ist.

PS: Ein Hinweis des Abo-Service für Abonnenten:
Es kann geschehen, daß Sie das letzte oder dieses Heft doppelt bekommen. Das hat einen einfachen Grund: Die ständig steigenden Abonnements zwangen langsam aber sicher unseren TI röchelnd in die Knie. (Verwalten Sie doch damit mal rund 11 000 Adressen!) Deshalb wurde der Abo-Service mit einem PC samt 20 Megabyte Festplatte ausgerüstet, der als erstes mal den Geist aufgab, sinnigerweise, als die ersten 4500 Abonnenten schon eingetippt waren. Einige Anschriften konnten wir retten, andere mußten neu eingegeben werden. So kann es geschehen, daß aus dem alten TI, dem ersten und auch dem 2. PC mal Adreßaufkleber doppelt kamen. Behaltet das Heft, verschenkt es, bitte nicht zurückschicken. Danke

BRIEFE

IN EIGENER SACHE

In der letzten Zeit wurden wir öfter gefregt, ob wir eigentlich Leserfregen nur am Telefon und en dieser Stelle hier beantworten. Nein, elle schriftlichen Fragen werden auch direkt von uns beantwortet. Leider liegen bei uns aber noch eine ganze Menge Fragen „auf Eis". Diese können wir nicht beantworten, weil nirgendwo ein Absender vermerkt war. Also, wer noch keine Antwort bekommen hat, schreibe uns bitte nochmal und bitte den kompletten Absender mit Anschrift auf dem Brief (nicht nur auf dem Umschlag!) vermerken. Letzteres gilt natürlich ganz allgemein. Briefe mit der Anschrift „Karl Schusselig, Deutschland" haben wenig Chancen anzukommen und werden von uns deshalb gar nicht erst weggeschickt.

Eprommer-Platine

In den Ausgaben 6/B5 und 7/85 hatten Sie einen Bericht über einen Eprom-Programmierer. Meine Frage ist, wo bekomme ich diese Platine mit dem Sockel? Außerdem: Kann man die Sockel im Elektronikladen kaufen, welche Nummer haben.die Eprom-Sockel?
Eine andere Frage: Wie kann man bei Extended-Basic den Befehl "Protected" übergehen oder gar löschen, um wieder Listen und Saven zu können?
In der Ausgabe 6/85 wurde das Modul Extended Basic II plus vorgestellt. Frage: Kann man die Grafik-Routinen nur mit Erweiterung benutzen oder kann man sie direkt vom Modul mit "CALL APESOFT" abrufen ohne die Apesoft-Grafik extra zu kaufen?
Carlos Gomez-Saez, Düsseldorf

Den Sockel für das Eprom können Sie in jedem Elektronikladen kaufen. Da es diese von verschiedenen Herstellern gibt, können wir Ihnen leider keine Nummer angeben, verlangen Sie einfach einen 2Bpoligen Nullkraftsockel. Die Fertigung und den Vertrieb der Platine hat für uns die Firma Elektronik-Service, Linning 37, 4044 Kaarst 2, übernommen. Weiteres entnehmen Sie bitte den Anzeigen dieser Firma. Protected können Sie wie folgt auflösen, was sich ja mittlerweile herumgesprochen haben dürfte: "CALL LOAD(-31931,0)". Um allerdings Call Load durchführen zu können, benötigen Sie eine Speichererweiterung. Das Modul Extended Basic II plus beinhaltet die kompletten Apesoft Grafik-Routinen. Mit CALL APESOFT werden diese in die Speichererweiterung aus dem Modul ausgeladen. Sie benötigen dazu also eine 32KByte Speichererweiterung, müssen aber die Apesoft-Grafik nicht extra kaufen.

Druckerprobleme

Ich habe Probleme mit meinem Drucker. Folgende Konfiguration benutze ich: TI 99/4A Konsole, Parallel TI Interface, Seikosha GP 250X Drucker. Dieser druckt nur 4 Buchstaben und rattert dann durch. Ich versuchte es mit dem Modul Datenverwaltung und Analyse, Texte- und Dateiverwaltung und auch Extended 8asic. Noch mehr Probleme gibt mir das Mini-Memory auf. Eventuell können Sie mir helfen.
Rene Mayer, Schwarzenbach

Leider geben Sie uns keine genaueren Hinweise, was beim „Durchrattern" exakt geschieht. Wir vermuten aber, daß bei Ihrem Kabel bzw. dem Anschluß des Interface eine der Leitungen "Busy" oder "Acknowlegde" nicht richtig verbunden ist. Diese Leitungen sind nämlich Ausgänge am Drucker und zeigen dem TI an, daß der Drucker nicht empfangsbereit ist. Ihre Fragen zum Mini-Memory scheinen sehr allgemeiner Natur zu sein. Dieses Modul ist ja eines der Module, die Maschinensprache für den TI 99/4A können. Die Anleitung zum Mini-Memory ist sehr dürftig. Wir können Ihnen hier nur empfehlen, sich dazu noch das Handbuch zum Editor/Assembler zu beschaffen. In diesem ist die Maschinensprache des TI 99/4A wesentlich ausführlicher beschrieben.

Speichererweiterung abschalten

Bei der Konfiguration Konsole, Minmem und Speichererweiterung kann man mit dem Befehl CALL LOAD (28706,0,0,0,0,0,0,0,0) die Speichererweiterung „unsichtbar" machen. Welche Möglichkeit gibt es nun, um in der Konfiguration Konsole plus Extended Basic und Speichererweiterung mit einem ähnlichen Befehl zu verhindern, daß ein Programm in Extended Basic in die Speichererweiterung ab >A000 aufwärts geschrieben wird, d.h., die Speichererweiterung unsichtbar wird?
Durch unterschiedliche Veröffentlichungen verwirrt, erhebt sich bei mir die Frage: Benötigt man zum Betrieb eines Druckers unbedingt die Expansions-Box (Minimem, Ext. Basic und 32K-Erweiterung vorhanden)?
NoNeme

Die Speichererweiterung kann unter Extended Basic wie folgt „abgeschaltet" werden: Sie geben folgendes Programm ein und starten es dann mit RUN.
100 CALL LOAD(-31868,0, 0):: RUN "DSK1.XXX"
Für XXX geben Sie den Programmnamen ein, welches Sie starten wollen, für den Kassettenrekorder muß es heißen: "CS1" statt "DSK1.XXX". Wichtig ist, daß nach dem Call Load unbedingt irgendein Programm eingeladen und gestartet wird, sonst gerät das Betriebssystem völlig durcheinander. "Einschalten" können Sie die Speichererweiterung übrigens wieder mit diesem Programm:
100 CALL LOAD(-3186B, 255,231):: RUN "CS1"
Zum Anschluß eines Druckers benötigen Sie an der TI 99/4A-Konsole ein Interface. Dieses muß in seinem Typ zum Interface des Druckers passen. Für den TI gibt es RS232- (auch als V24 bezeichnet) und Centronics-Interface (Parallel-Schnittstelle). Beide sind sowohl als Karte für die Peripheriebox wie auch als "stand-alone"-Ausführungen erhältlich, die direkt seitlich an der Konsole angesteckt werden. Schlußendlich gibt es auch noch zwei verschiedene 32KByte-Speichererweiterungen mit eingebautem Centronics-Interface, die seitlich an den TI angesteckt werden.

Zum Kassettenrekorder

Ich konnte Ihre Kassette N. 5 wie auch die Rückseite einer anderen, bespielt gekauften Kassette nicht laden. In vager Erinnerung eines Tips in der TI-REVUE 6 zog ich den "Save"-Stecker aus der Mikrophonbuchse des Kassettengerätes, danach OLD CS1 usw. ganz normal und siehe da: DATA OK. Können Sie mir erklären, warum das so geht?
Peter Bigler, Muri

Der Kassetteneingang des TI 99/4A benötigt natürlich ein einigermaßen gutes Signal vom Kassettenrekorder, d.h. das Signal darf nicht zu viel Rauschen, Brummen oder sonstige Störimpulse aufweisen. Je nach interner Beschaltung des Kassettenrekorders kann es sein, daß durch Mikrophon und Kopfhöreranschluß die Masse doppelt verbunden ist, was zu den Störeffekten einer Masseschleife führen kann. Ebenso kann der doppelte Anschluß im Kassettenrekorder zu einer niedrigen Ausgangsspannung führen. Das alles ist aber vom Kassettenrekorder abhängig und kann schlußendlich mit nicht auf diesem Gerät bespielten Kassetten eine zu hohe Störung des Signals ergeben.

Neuer TI-Computer?

Eigentlich wollte ich den Trick aus Heft 7/B5 von Franz Becker ausprobieren. Aber ich kam auf etwas sehr seltsames. Bitte probieren Sie folgendes:
1 I (Ctrl U drücken bis Zeile voll ist)
Dann Enter drücken und NUM 1 eingeben. Der ganze Bildschirm wird voll mit "RANDOMIZE".
Kurz danach wird der Bildschirm gelöscht. Nun machen Sie folgendes.
4mal 1 drücken: RANDOMIZE erscheint wieder.
1mal 2 drücken: Bildschirm wird gelöscht.
9mal 3 drücken: Rechts und links orange Balken, in der Mitte orange und grüne Streifen.

Leserbriefe und Fragen sind uns stets willkommen. Wir beantworten sie entweder direkt oder auf der Leserbriefseite

BRIEFE

14mal 4 drücken: Bildschirm wird orange mit blauen Punkten.
1mal 5 drücken: 2 große Strings erscheinen auf dem Bildschirm.
Es gibt unendlich viele Tastenkombinationen. Das ganze geht auch mit anderen Control -Zeichen. Der Hinweis von Franz Becker ging bei mir nicht. Statt der Grauabstufungen kamen nur Basic-Anweisungen. Gibt es dafür eine Erklärung oder ist mein TI defekt?
Weiter habe ich in einer anderen Zeitschrift einen Bericht gelesen, daß in den Staaten ein verbesserter TI mit mehr Speicherplatz, besserer Grafik und höherer Geschwindigkeit herauskommen soll. Haben Sie nähere Informationen darüber oder ist das nur ein scherzloser Einfall eines Redakteurs gewesen?
Steffen Keller, Merklingen

Solche Tricks, die das Betriebssystem in irgendeiner Form überlisten, müssen nicht auf allen TI 99/4A gleich funktionieren. Einmal gibt es verschiedene Betriebssysteme und zum anderen werden bei solchen Sachen manchmal auch die internen Bausteine (IC's) in nicht definierten Zuständen gehalten, was je nach Produktionsserie verschiedene Auswirkungen haben kann. Defekt ist Ihr TI aber deshalb noch lange nicht. Die Frage nach dem neuen TI-Computer kann zur Zeit recht einfach beantwortet werden: Es gibt ihn nicht. Richtig ist, daß sich seit der Produktionseinstellung des TI 99/4A hartnäckig Gerüchte (vor allem in den USA) halten, nach denen ein neuer TI 99/4A-kompatibler Computer auf den Markt kommen soll. Es gibt in den USA auch schon einige Prototypen bei kleineren Herstellern. Das momentan aussichtsreichste Projekt sieht folgendermaßen aus: TMS 9995 CPU, 128K-Byte Speicher, MSX II-Video-Prozessor (80 Zeichen, mehrere Spriteebenen, 256 Farben und bis zu 128KByte VDP-Ram). Dieser Computer soll voll softwarekompatibel zum TI 99/4A sein, d.h. die bisherigen Module sollen laufen. Auch die bisherige P-Box kann wohl mit einer geänderten Treiberkarte weiter betrieben werden. Probleme gibt es aber, wie aus amerikanischen Händlerkreisen zu erfahren war, mit 2 Dingen: Einmal werden von Texas Instruments Lizenzrechte benötigt und zum anderen scheint die Finanzierung der Produktion noch nicht gesichert, so daß es heute noch nicht feststeht, ob dieser Computer überhaupt produziert werden kann. Und dann ist bei der „geradlinigen" Geschäftspolitik ja auch Texas Instruments selber durchaus für eine Überraschung gut. Dennoch wollen wir hier nicht aufgrund irgendwelcher Gerüchte falsche Hoffnungen wecken. Wir glauben an den neuen 99er Computer erst, wenn er bei uns auf dem Redaktionstisch steht.

Reset-Schalter

Ich habe folgende Fragen an Sie:
1. Wo befindet sich am Modulport der Resetausgang?
2. Gibt es schon Schnittstellen zum Selberbauen?
3. Wenn ich im Extended Basic CALL PEEK(-2622,A) eingebe, hängt sich mein Computer auf. Woher kommt das?
Michael Bursesch, Elmshorn

Der Resetanschluß ist Pin 1 am Modulport. Verbinden Sie diesen über einen 100 Ohm-Widerstand und einen Taster mit dem Anschluß für -5V (Pin 29) Pin 1 an diesem Stekker befindet sich unten links, Pin 2 darüber. Rechts neben Pin 1 liegt Pin 3 usw. Schnittstellen zum Selberbauen sind uns bisher nicht bekannt, aber wir arbeiten an einer. In diesem Zusammenhang hätten wir eine Frage an unsere Leser: Wer uns dafür die Platine entwerfen will, bzw. kann, möchte sich bitte bei der Redaktion melden. Mit CALL PEEK(2622,A) verändern Sie die Grom-Adresse. Da der TI 99/4A danach dann nicht mehr an der gleichen Stelle wie vor dem Call Peek ist, kommt es zu der Fehlfunktion. Es wird im Betriebssystem quasi ein Byte im Programm übersprungen.

Die Club-Ecke

In der letzten Zeit haben einige Clubs es vorgezogen, statt sich hier auf der Clubseite vorzustellen, in den Kleinanzeigen auf sich aufmerksam zu machen. Dies, obwohl doch hier jedem Club wesentlich mehr Platz zur Verfügung steht. Sicherlich hatten wir bei noch zweimonatlicher Erscheinungsweise einige Male eine kleine Warteliste für die Clubseite. Diese ist aber nun durch das monatliche Erscheinen abgebaut und so ist dieser Platz doch besser für alle Clubnachrichten geeignet als die Kleinanzeigen. Dabei wollen wir ja auch nicht nur allgemeine Clubvorstellungen veröffentlichen, sondern sind auch an Erfahrungsberichten von der täglichen Clubarbeit interessiert. Andere Clubs können doch nur von den Erfahrungen der schon längere Zeit bestehenden Clubs lernen. Schlechte Erfahrungen sind da genauso nützlich wie positive.

Beginnen wollen wir heute wieder einmal mit einem Aufruf zur Clubgründung:
Wer hätte Lust, bei einem überregionalen TI 99/4A Club mitzumachen. Der jährliche Clubbeitrag beträgt 10,– DM. Ein für die Mitglieder frei verfügbarer Software-Pool soll aufgebaut werden (nur selbstgeschriebene Software). Ca. alle 2 Monate soll ein von den Mitgliedern gestaltetes Magazin auf Kassette herausgebracht werden. Nähere Informationen erhaltet ihr gegen einen frankierten Rückumschlag. Wenn ihr eigene Ideen zur Gestaltung des Clubs habt, schreibt sie bitte:
Thomas Klein, Am Ring 18, 6759 Wolfstein/Pfalz

Die nächste Clubvorstellung erreichte uns aus dem Raum Frankfurt/Darmstadt:
Wir sind eine Gruppe von derzeit 4 Personen, die sich in ihrer Freizeit mit Programmierung und Hardware des TI 99/4A beschäftigen.
Da es im Großraum Frankfurt/Darmstadt keinen Club gibt, der sich ausschließlich dem TI 99/4A widmet, haben wir den Entschluß gefaßt, diesen Mangel abzustellen.
Mit dieser Vorstellung hoffen wir, auch andere TI-User in diesem Raum anzusprechen. Wir sehen den Sinn eines solchen Clubs u.a. in dem Gedankenaustausch, der gemeinsamen Entwicklung von Hard- und Software sowie der gegenseitigen Hilfe und dem persönlichen Kontakt. Ein Mitgliedsbeitrag ist nicht geplant, deshalb auch vorerst keine Clubzeitschrift. Da wir aber nicht wissen, wie sich die Sache anläßt, kann sich noch vieles an der Struktur ändern. Falls jemand Interesse hat, kann er (sie) sich an folgende Adresse wenden:
Christopher Winter, Rembrücker Weg 44, 6053 Obertshausen.

Ein weiterer Club ist in Mönchengladbach beheimatet und sucht dringend Mitglieder:
Wir, der Technomania-Club, suchen verzweifelt TI-User im Raum Mönchengladbach. Trotz mehrerer Anzeigen sind wir bis jetzt erst 2 Leute. Sollten es mehr werden, so haben wir einiges vor: der Schwerpunkt des Clubs sollte auf dem Software-Sektor liegen (TI- und X-Basic). Auch eine Clubzeitschrift und eine Programmzentrale (Tausch) soll entstehen. Kontakte mit schon bestehenden, anderen TI-Clubs in Deutschland werden gerne geknüpft (z.B. überregionaler Softwareaustausch). Ein regelmäßiges Clubtreffen sollte auch stattfinden. Wann, wo und wie dies alles passiert, wird bei ausreichender Mitgliederzahl ausgemacht. Interessenten (auch Clubs) bitte melden, am besten mit Telefonnummer bei:
Marco Scheloske, Von-Groote-Str. 154, 4050 Mönchengladbach 1 oder bei:
Heinz-Josef Eckers, Hermann-Hesse-Str. 7, 4050 Mönchengladbach 1.

Zum Schluß für heute noch eine Meldung vom TI 99er Workshop, in deren Reihen wohl ein Werbefachmann sitzen muß, denn dort hat man wohl begriffen, wie man es machen muß, um immer wieder in die Clubseite zu kommen:
Da die Nachfrage zur Mitarbeit in unserem Workshop, nicht zuletzt durch die Veröffentlichungen in der TI-REVUE sehr groß geworden ist, haben wir unsere Kontaktadressen auf verschiedene Themenbereiche aufgeteilt. Für Hardwarefragen aller Art ist ab sofort folgende Kontaktadresse eingerichtet: TI 99 Workshop Rheinland, c/o Mike Dean, Neußer Str. 47, 5000 Köln 1. Für allgemeine Anfragen und Software ist weiterhin zuständig:
Mike Heuser, Karl-Marx-Allee 47, 5000 Köln 71.

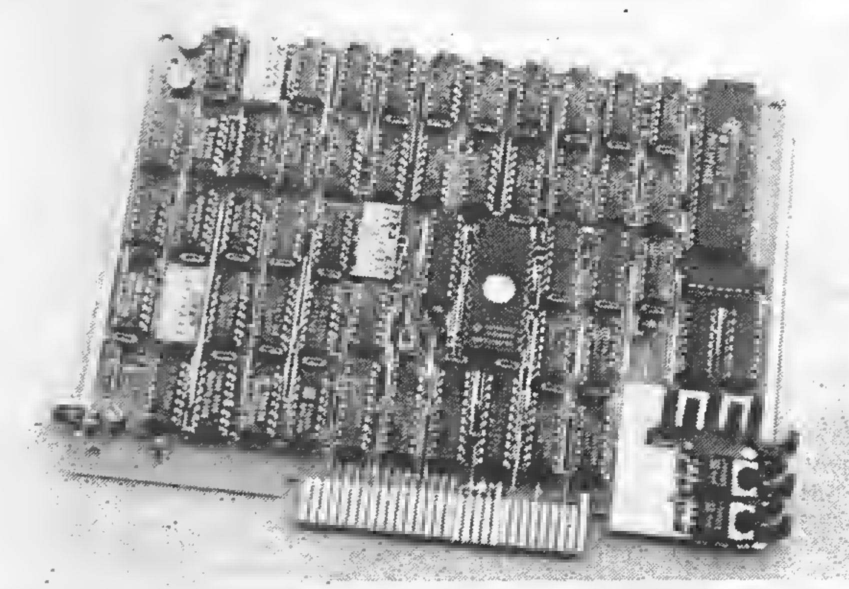
Von GPL &

GPL (Graphics Programming Language), wie Texas Instruments das bezeichnet, das war wohl das große Geheimnis des TI 99/4A. Bei der Konstruktion des TI 99/4A sah Texas Instruments spezielle Bausteine, die Grom's vor. Diese Bausteine arbeiten in der sogenannten „Memory-mapped" Technik, d.h. sie beinhalten einen internen Zähler, der durch gezieltes Schreiben über die Datenleitungen auf bestimmte Werte gesetzt werden kann. Dieser Wert wird dann als Adresse für ein ROM (nur Lesespeicher) verwendet. Und der Inhalt der damit adressierten Speicherstelle in dem ROM kann anschließend ausgelesen werden. Bei jedem Lesevorgang erhöht sich der Zählerstand automatisch um 1, so daß nacheinander der Inhalt der ROM's gelesen werden kann, ohne daß jedesmal neu die Adresse in den Zähler geschrieben werden muß.
Nun sah Texas Instruments im Betriebssystem eine spezielle Programmiersprache namens GPL vor, deren Programme nur in diesen GROM's enthalten sein müssen. Diese Sprache ist sehr prozessornah und ähnelt damit der Maschinensprache (Assembler) des TMS 9900 Prozessors. Damit wollte Texas Instruments die Software-Entwicklung (Module) von Fremdfirmen wohl unterbinden bzw. von Lizenzen durch Texas Instruments abhängig machen. Dieses Vorhaben ist ja auch lange gelungen, jedoch in letzter Zeit durch die entsprechenden Veröffentlichungen völlig unterlaufen worden. Wir haben in der TI-REVUE eine Modulplatine vorgestellt, auf der durch ein paar TTL-IC's und EPROMS diese GROM's simuliert werden. Damit, und mit der Veröffentlichung aller GPL-Befehle in dem Buch „TI 99/4A Intern", war der Anfang gemacht, auch diese Programmiersprache des TI 99/4A allgemein zugänglich zu machen.
Nun ist es ja aber sehr zeitraubend, ein Programm in GPL zu schreiben und zur Fehlersuche jeweils in ein EPROM zu brennen. Also war die Entwicklung von GRAM's notwendig. Diese konnten und wurden sehr einfach aus der Modulplatine entwickelt, aber es ist nun auch schon eine „große" Lösung, eine GRAM-Karte für die P-Box angekündigt. Damit ist nun, zusammen mit den erwähnten GPL-Assemblern, ein komplet-

GRAMS & GROMS

Eigentlich sollte dieser Bericht ein Test von zwei neu auf den Markt gekommenen GPL-Assemblern werden. Schon bald wurde allerdings klar, daß eine Besprechung nur dieser Programme, zur jetzigen Zeit wenig Sinn hat, sondern daß zuerst einmal über die Möglichkeiten, die GPL bietet, gesprochen werden muß.

tes Entwicklungssystem für GPL-Programme vorhanden. Es können also einfach neue, eigene Module entwickelt werden, die auch wirklich die ganzen Möglichkeiten des TI 99/4A ausnützen. Erfahrene Maschinenprogrammierer werden dem nun entgegen halten, daß das in Assembler doch auch geht. Da ist zum Beispiel auf den umständlichen Zugriff auf den Kassettenrekorder hinzuweisen; in GPL geht das viel einfacher. Und dann ist da noch das wichtigste: Assemblerprogramme



Ein Prototyp der im Text erwähnten GRAM-Karte für die P-Box

benötigen die Speichererweiterung (abgesehen von den nicht auf jeden TI99/4A benutzbaren ROM-Modulen). GPL-Module aber können, sofern sie nicht aus anderen Gründen eine Speichererweiterung benötigen, nur auf der Konsole arbeiten!
Auch können in GPL einfach Basic-Erweiterungen geschrieben werden. Kommen wir nun wieder auf die GPL-Assembler zurück. In der Bundesrepublik Deutschland werden momentan 2 verschiedene Assembler angeboten: Einmal das GPL-Assembler/Disassembler-Paket der Firma Reis, Bullay, zum Preis von 149,– DM und zum anderen ein GPL-Assembler von der Firma Elektronik Service, Kaarst zum Preis von 99,– DM. Beide Assembler sind unterschiedlich, bleiben wir aber erst einmal beim Disassembler. Dieser erlaubt die Ausgabe des Listings auf Drucker, Diskette oder Bildschirm. Wahlweise kann das Listing die byteweise Ausgabe des hexadezimalen Werts, ein Hexdump oder eben das disassemblierte in den GPL-Mnemonics dargestellte Programm enthalten. Dabei können GPL-Programme nicht nur aus den GROM's, sondern auch aus VDP-RAM oder CPU-RAM disassembliert werden. Dieser Zusatz ist sehr nützlich, wenn man auch ohne GRAM's z.B. GPL-Programme nur zum Disassemblieren von Diskette in den Speicher lädt. Ebenfalls wichtig bei Disassembler ist die Funktion, daß nach Aufruf eines Unterprogramms mit dem GPL-Befehl CALL ein Byte übersprungen werden kann. Dieses wird ja sonst durch den GPL-Befehl FETCH gelesen, hat mit dem Programm selber ja nichts zu tun und würde sonst den Disassembler völlig durcheinander bringen.
Beide Assembler benötigen als Quellcode eine Textdatei im DIS/VAR 80 Format, wie sie mittels des Editors vom Editor/Assembler-Modul und des TI-Writers erstellt werden können. Die Ausgabe erfolgt beim Assembler der Firma Reis wahlweise als program-image oder als DIS/FIX 80 Datei. Der Assembler des Elektronik-Service erstellt dagegen Tagged-Object-Code, wie es auch das Editor/Assembler-Modul macht.
Beide GPL-Programme in absoluter Adressierung erstellen.
Relocatibler Object-Code ist ja mangels eines geeigneten Laders auch nicht sinnvoll.
Ebenfalls gemeinsam ist beiden GPL-Assemblern, daß sie alle GPL-Befehle umwandeln, allerdings ist bei demjenigen der Firma Reis beim FMT-Befehl eine Einschränkung zu machen: Hier müssen die Unterbefehle mittels der Anweisung BYTE eingegeben werden.
Hinsichtlich der Pseudo-Befehle bzw. Direktiven kennt der GPL-Assembler der Firma Elektronik-Service aber erheblich mehr Funktionen, die das Arbeiten erleichtern.
Die Verwendung von Labels ist bei beiden Assemblern möglich, allerdings dürfen die Labels bei dem der Firma Reis nur für Sprung- bzw. Verzweigungsadressen benutzt werden. Berechnungen im Adressfeld sind nur bei dem GPL-Assembler der Firma Elektronik-Service möglich.
Will man ein Fazit über die beiden Programme ziehen, so ist zu sagen, daß der GPL-Assembler der Firma Elektronik-Service sehr komfortabel ist, in seiner Ausführung dem normalen TMS 9900 Assembler des TI 99/4A ähnelt. Er läuft auch auf dem Editor-Assemblermodul und wird einfach anstelle des normalen Assemblers gestartet. Das Paket der Firma Reis umfaßt ja nicht nur einen GPL-Assembler, sondern auch einen Disassembler, welches schlußendlich zu berücksichtigen ist.
Dieser Assembler ist nicht ganz so komfortabel, dafür ist der Disassembler bisher in dieser Form einzig auf dem Markt. Benötigt wird hier das Editor/Assemblermodul oder das Extended Basic. Beide Assembler benötigen zu dem jeweiligen Modul noch Disk-Laufwerk und Speichererweiterung. Ein Drucker ist ganz bestimmt sinnvoll.
Beide Assembler wurden mit kurzen Beispielen auf ihre Richtigkeit überprüft und zeigten keine Fehler. Allerdings muß hier angemerkt werden, daß eine Prüfung auf vollständige Fehlerfreiheit aus Gründen des Aufwandes nicht durchgeführt werden konnte. Zu guter Letzt bleibt noch anzumerken, daß alle dieser Programme in TMS 9900 Assembler geschrieben sind und entsprechend schnell arbeiten.

Heiner Martin

32 muntere Kilos für ganze 75 D-Märker!

Speichererweiterung für den TI ist ein immer wieder auftauchendes Thema. Das mag daran liegen, daß auf dem Weg zur Assemblerprogrammierung immer die Hürde Speichererweiterung zu nehmen ist. Es sei denn, man hat ein Minimemory und beschränkt sich auf die darin enthaltenen 4 kB Speicherraum. Die 32 kB-Erweiterung war bisher aber auch eine kostspielige Anschaffung. Unter 300,– DM war wohl kaum etwas zu erhalten. Inzwischen hat sich der Preis von Speicher-IC's aber erheblich verringert. Immerhin sind die 5 IC's dieser Schaltung für insgesamt weniger als 75,– DM erhältlich.

Technische Daten:	
Speicher	statische RAM 8K (8192*8bit)
Speicherkapazität	32 kB (32768 Byte)
Decodierter	>2000...>3FFF (8KB)
Adressbereich	>A000...>FFFF (24KB)
Spannungsversorgung	5 V aus der Konsole
Standby-Spannung	Batterie oder Akku
	2,4 bis 3,6 V
Stromaufnahme	Betrieb: 15...30 mA
	Standby: 3...10 uA!!
Masse	ca. 105*65*15 mm

Schaltungsbeschreibung

Die Schaltung setzt sich aus Speicher, Decoder und Standby-Versorgung zusammen.
Als Speicher werden 4 statische RAM 6264 LP 15 verwendet.
Statische RAM sind ja bekanntlich leichter zu handhaben als dynamische RAM, wodurch außer dem Decoder keine weiteren Bausteine für den Betrieb erforderlich sind.

Puffer für Adress-, Daten- und Steuerleitungen wurden in dieser Schaltung bewußt weggelassen, da die Adress- und Steuerleitungen in der Konsole vor dem Bus-Ausgang gepuffert sind. Die Datenleitungen der RAM sind nur bei read/write-Zyklen durchgeschaltet, ansonsten hochohmig.

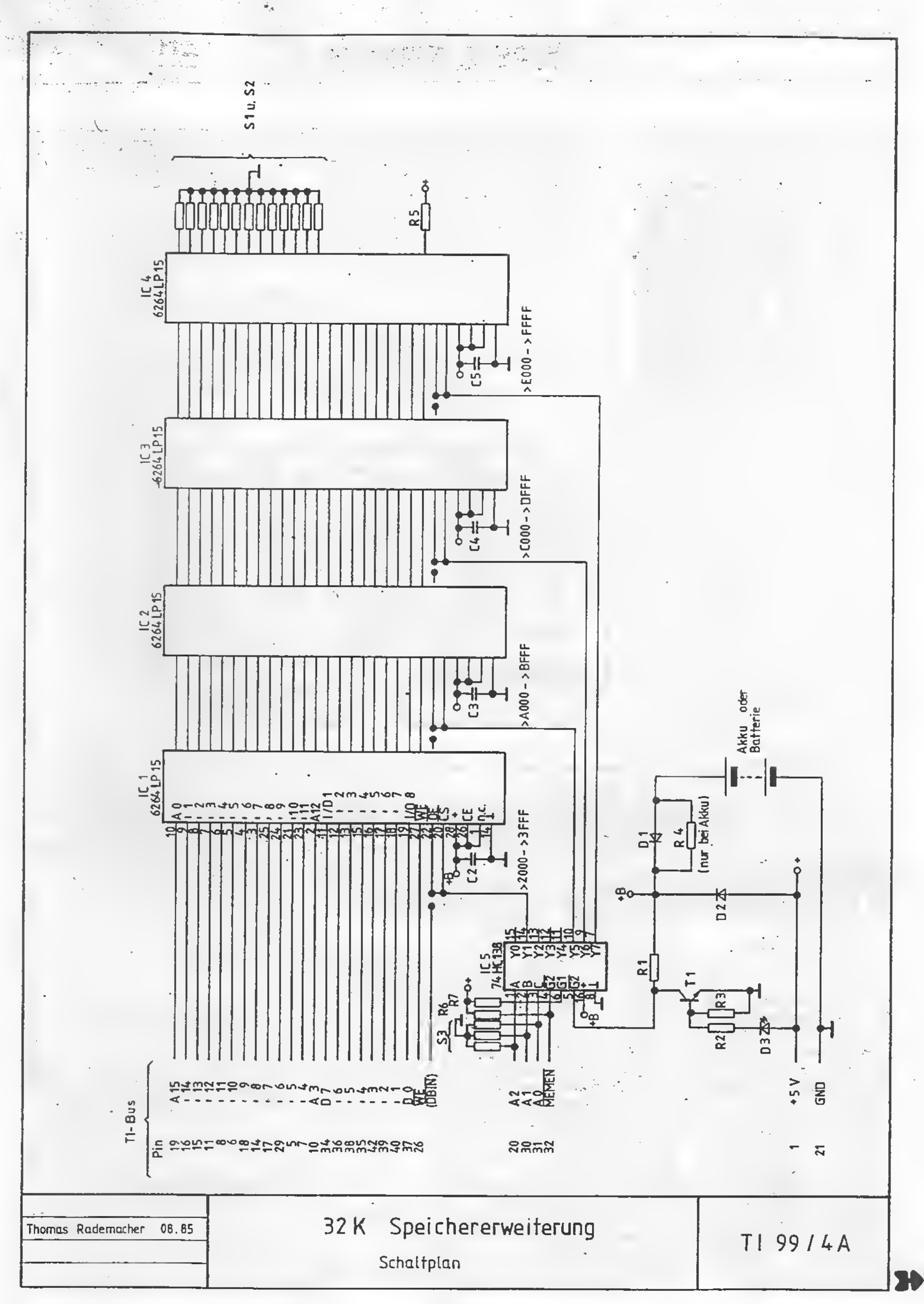
Die Zusammenschaltung der RAM und der Anschluß an den TI-Bus ist wohl kaum anders möglich. Als Decoder wird ein 74 HC 138 verwendet. Dieser zeichnet sich gegenüber dem 74 LS 138 durch eine erheblich geringere Stromaufnahme aus (wichtig für den Standby-Betrieb). Der Decoder bildet aus den Adressleitungen A0, A1 und A2, verknüpft mit MEMEN, die 8 kB-Bereiche für die einzelnen RAM.
Die Standby-Schaltung soll die RAM auch nach Ausschalten der Konsole weiter mit Spannung versorgen, so daß gespeicherte Daten auch über Wochen und Monate erhalten bleiben und weiter verwendet werden können.
Für den Standby-Betrieb benötigen die RAM's laut Hersteller mindestens 2 V. Daher muß die Batteriespannung mindestens 2,4 V sein, da an der Entkoppeldiode D1 (Germaniumdiode) ca. 0,3 bis 0,4 V verloren gehen. Andererseits sollte die Batteriespannung 4,0 V nicht überschreiten, denn sie muß unterhalb des Wertes der Betriebsspannung bleiben (die 5 V verringern sich noch um etwa 0,6 V, die an D2 abfallen). Andernfalls wird die Batterie trotz anliegender Betriebsspannung entladen.
Mit R1, R2, R3, D3 und T1 wird der G2-Eingang des Decoders auf 'HIGH' gelegt, sobald die 5 V von der Konsole fehlen. Der Decoder wird hierdurch blockiert und sperrt über die CS-Eingänge die RAM's. Dadurch werden einerseits Zugriffe ins RAM unterdrückt, andererseits die Stromaufnahme der RAM's erheblich gesenkt. Bei meinen Speichererweiterungen habe ich im Standby-Betrieb eine Stromaufnahme von etwa 5 uA gemessen (Batterie 2,4 V). Damit dürfte eine Batterie ganz schön lange halten.
Falls ein Akku, statt der Batterie eingesetzt wird, muß R1 eingelötet werden. Über R1 wird der Akku nachgeladen. Der Widerstandswert ist abhängig von der Akkukapazität. Er sollte so dimensioniert werden, daß ein Ladestrom von 1/10 der Akkukapazität fließt.
Die Pull-Up-Widerstände R5 und R6 sowie die Pull-Down-Widerstände S1 bis S3 sind nicht unbedingt erforderlich, wurden jedoch vorsichtshalber eingeplant.

Bestückung und Anschluß

Die Bestückung geht aus dem Bestückungsplan hervor. Die Anschlußpunkte der Platine sind für eine 34polige Pfostenfeld-Steckverbindung mit Flachbandkabel vorgesehen. Das Kabel kann aber auch direkt angelötet werden. An den Stecker zum TI-Bus messen die einzelnen Adern des Kabels, wie aus dem Schalt- und Bestückungsplan hervorgeht, angelötet werden.
Hier noch ein Tip zum Einbau eines durchgereichten Busses in die Speichererweiterung. In Elektronik-Fachgeschäften gibt es eine Platine (RTO Nr. 6602 ca. 10,– DM) womit sich der Bus einfach verlängern läßt. An eine Stirnfläche der Platine ist, wie aus Bild 1 ersichtlich, der 2x22polige Direktstecker anzulöten. In der Mitte der Platine sind alle Leitungen an Lötpunkten zugänglich. Die Platinenlänge ist 65 mm. Sie paßt also genau quer unter oder über die Speicherplatine.
Erwähnenswert ist vielleicht noch, daß ein Einbau der Speicherplatine in die Konsole möglich ist. Allerdings ist das Anlöten der 34 Leitungen auf der Rechnerplatine eine haarsträubende Angelegenheit und sollte wirklich nur von richtigen Elektronik-Freaks durchgeführt werden. Immerhin haben bei bisher 3 Versuchen alle Rechner diesen Eingriff „lebend" überstanden.

Test

Nach Fertigstellung der Speichererweiterung kommt der große Moment:

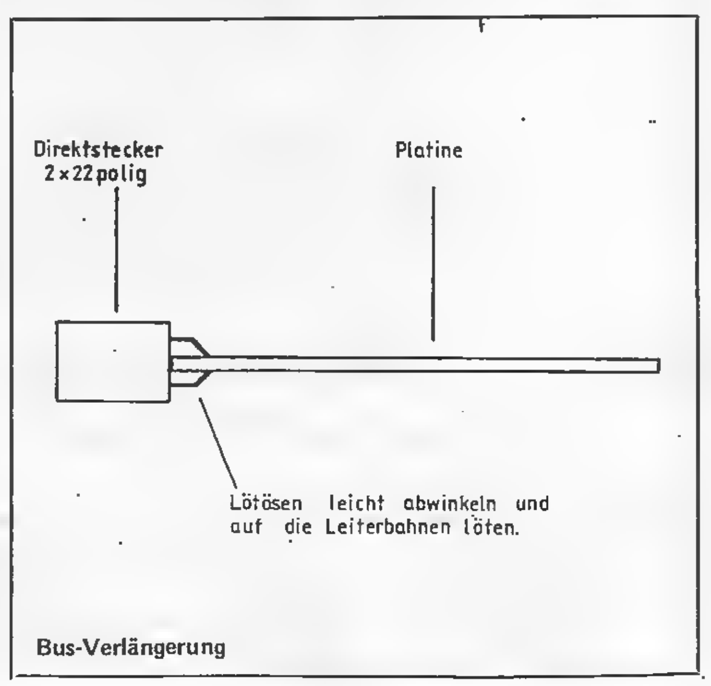

Thomas Rademacher 08.85
32 K Speichererweiterung
Schaltplan
TI 99 / 4 A
IC 1 6264 LP15
IC 2 6264 LP15
IC 3 6264 LP15
IC 4 6264 LP15
IC 5 74 HC 138
S1 u. S2
S3
R5
R6
R7
R1
R2
R3
R 4 (nur bei Akku)
T1
D1
D2
D3
C2
C3
C4
C5
>2000->3FFF
>A000->BFFF
>C000->DFFF
>E000->FFFF
Akku oder Batterie
TI-Bus
Pin
+5 V
GND
MEMEN
(DBIN)

das erste Einschalten. Nach Erscheinen des Titelbildes und Auswahl des XBasic-Modus, kann als erster Test 'SIZE' eingegeben werden. Der Computer müßte sich daraufhin mit '13928 BYTES OF STACK FREE' '24488 BYTES OF PROGRAM SPACE FREE' melden.

Ist soweit alles okay, sollten alle Bereiche der Speicherkarte durch Ein- und Auslesen von Daten mit CALL LOAD(...) und CALL PEEK(...) getestet werden (vorher muß CALL INIT eingegeben werden). Wenn alle Tests erfolgreich verlaufen sind, hat Ihr TI jetzt 32 kB mehr Speicherraum. Haben Sie auch eine Standby-Batterie eingebaut, läßt sich ein eingeladenes Basic-Programm nach Aus- und Einschalten der Konsole wieder zugänglich machen, indem vor dem Ausschalten mit CALL PEEK (-31952,A1, A2,B1,B2) die Daten ausgelesen und notiert werden. Nach dem Einschalten der Konsole müssen diese Daten dann mit CALL LOAD (-31952, A1,A2,B1,B2," ",-25,0) wieder eingegeben werden. Danach kann man RUN oder LIST eingeben und abwarten was sich tut.

Thomas Rademacher

Anm.d. Red.: Statische 8 k-RAM's sind leider am TI 99/4A nicht immer ohne Probleme zu betreiben. Dies liegt daran, daß beim TI 99 die Adressleitungen geändert werden, ohne daß der Speicherbaustein deselektiert wird (CE und OE bleiben aktiv). Leider können hier keine weiteren Empfehlungen hinsichtlich der speziellen Type des RAM-Bausteines gegeben werden, da dieses Verhalten auch schon bei unterschiedlichen Serien eines Herstellers auftraten. Die überwiegende Mehrzahl der IC's arbeitet aber korrekt. Im Schaltbild sind OE und CE verbunden. Dies kann ebenfalls bei Ersatztypen zum 6264 zu Fehlfunktionen führen. Hier ist dann OE mit DBIN zu verbinden, wobei das Signal DBIN zusätzlich zu invertieren ist.



Bus-Verlängerung

TIPS & TRICKS

Stückliste

Halbleiter

IC 1...IC 4	6264 LP 15
IC 5	74 HC 138
T 1	BC 547 C
D 1	Germaniumdiode z.B. AA 119
D 2	Siliziumdiode z.B. 1 N 4148
D 3	Leuchtdiode rot

Kondensatoren

C 1	Tantalelko 47 µF/16 V
C 2...C 5	100 nF

Widerstände

R 1	270 k
R 2	220
R 3	68
* R 4	nur bei Akku erforderlich
* R 5	10 k
* R 6	10 k
R 7	10 k
* S 1 und S 2	Widerstandsnetzwerk 7x100 k mit gemeinsamer Masse
* S 3	Widerstandsnetzwerk 3x100 k mit gemeinsamer Masse

(für S 1 bis S 3 können auch Einzelwiderstände verwendet werden)

Sonstiges

*	4 IC-Fassungen 28polig
*	1 IC-Fassung 16polig
*	Batterie oder Akku 2,4 — 3,6 V
*	1 Pfostenfeldsteckverbinder
	1 Direktstecker 2x22polig (f.d. TI-Bus)
*	1 Gehäuse

Bauteile mit "*" sind nur bei Bedarf erforderlich

Die nächste TI-REVUE erscheint am 27. Dezember

DFÜ-Tips

Heute möchte ich über den Filetransfer vom TI zu einer MB berichten. Wir wollen einen Text, der wie folgt aussieht, zu einer MB senden:

V 1.0 Achtung TI-User
Verkaufe TI-Basic und TI-XBasic Programme.
Suche E/A oder MM Programme.
Liste anfordern unter Tel.Nr.:..:.........

Dieser Text sollte in einem Internal/Variablen 80 Format sein, da ein solcher File von dem Auto Log-on direkt übertragen werden kann.
Der Auto Log-on File wird automatisch abgefragt und falls gefunden, sofort übertragen. Dies kann beim Auswählen des 3. Menüpunktes von der Hauptwalliste des TE II geschehen. Falls man in einer MB momentan tätig ist, muß man den TI abschalten und wieder einschalten, damit der Auto Log-on File übertragen werden kann.
Wir gehen davon aus, daß der File gerade übertragen wurde und die MB diesen Text nicht angenommen hat. Dies hat folgenden Grund. Das Modul TE II braucht am Anfang eines Satzes einen Marker, damit das Modul „weiß", daß der Satz gesendet werden kann, damit das Modul „weiß", wann der Satz zu Ende ist, braucht es ebenfalls einen Marker. Dies würde so aussehen:

```
V 1.1
 ——Startmarker (Sende Text)
 /                     Endmarker (Textende) ——\
/                                              \
I                                               \
1  Achtung TI-User                              CR
1  Verkaufe TI-Basic und TI-XBasic Programme.   CR
I  Suche E/A oder MM Programme.                 CR
1  Liste anfordern unter Tel.Nr.:............ . CR
```

Jetzt kann es vorkommen, daß eine MB nach jeder Zeile, die sie empfängt, ein Steuerzeichen sendet. Um dieses Steuerzeichen zu empfangen, muß der TI einige Zeit Pause machen, um dann wieder eine Zeile zu senden oder dieselbe Zeile zu wiederholen.
Dieses Problem können wir so lösen:
Dazu sehen wir uns das Beispiel V 1.1 an. Wir haben dem TE II „gesagt", wann es einen Text senden soll und wann der Text bzw. Zeile zu Ende ist.
Jetzt müssen wir eine Warteschleife schaffen, damit eventuelle Steuerzeichen empfangen werden können. Dies würde so aussehen:

```
V I.2

1  Achtung TI-User                              CR
2  * <ASC II Code 42 siehe Erläuterung >
I  Verkaufe TI-Basic und TI-XBasic Programme.   CR
2  *
1  Suche E/A oder MM Programme.                 CR
2  *
1  Liste anfordern unter Tel.Nr.:............ . CR
2  *
```

Nun taucht im Beispiel V I.2 eine 2 und ein Asterix> ASCI II Code 42< auf.
Hier die Erläuterung:
Die >2< bedeutet für das Modul >Warte auf ein Steuerzeichen der Gegenstelle<. Wobei ich hier noch kurz erwähnen möchte, daß die MB immer der >Master< ist und der Anrufer der >Sklave<. D.h. der Anrufer macht im Prinzip nur das was die MB von ihm verlangt.
Weiter mit der >2< und dem Asterix. Das Modul wartet, falls wir eine >2< in einer Zeile haben, auf ein Steuerzeichen der MB. Das Modul „weiß" an dieser Stelle nicht, wie lange es warten soll. Dazu dient uns das Asterix.
Zum besseren Verständnis, eine Tabelle aus der die Verzögerungszeit zu ersehen ist.

ASC II Code	Zeichen	Realzeit in sec.
6	REDO	0,1
12	PROC'D	0,2
18	" "	0,3
24	" "	0,4
30	" "	0,5
36	S	0,6
42	*	0,7
48	0	0,8
54	6	0,9
60	<	1,0
66	B	1,1
72	H	1,2
78	N	1,3

Wir sehen, je höher der ASCII Code, desto länger ist die Wartezeit des Moduls.
Noch eine kurze Anmerkung zur Wartezeit:
Es wäre ratsam, daß die Wartezeit nicht über 42 sec. geht, da viele MB's ein relativ kurzes >timeout< haben. Der ASC II Code >*< hat sich hierbei gut bewährt.
Es sei zu hoffen, daß jetzt viele TI-User mit dem TE II Modul besser zurecht kommen als vorher.

Martin Miosga

Strings mit Charakter

Wie oft habe ich mich schon geärgert, wenn in einem Definitionsprogramm für Charakter zwar die definierten Zeichen „gesaved" werden aber nicht direkt in ein Programm eingebaut werden können! Dieses Programm habe ich einmal angepackt und bin zu folgender Lösung für den TI-99 plus Diskettenlaufwerk und Extended Basic (eventuell mit 32 K-Erweiterung) gekommen.

Jeder Computer, der als Programmiersprache BASIC besitzt, verfügt über sogenannte Basic-Token. Das sind alle Befehle, die der Computer im Basic hat (z.B.: PRINT, RUN . . .).
Diese kann man alle über CHR$ ansprechen und auf Disk saven. Um diese auf Diskette saven und später auch bequem einlesen zu können, muß man am besten ein Format für den OPEN-Befehl wählen, das dies erlaubt. Vom Diskettenlaufwerk steht hier der Befehl "MERGE" zur Verfügung, der es erlaubt Programmzeilen in den Arbeitsspeicher zu laden, ohne daß das im Speicher befindliche Programm gelöscht wird (höchstens, wenn sich die Zeilennummern schneiden!).

Nun zum Aufbau des Merge-Formats:

Dateien im Merge-Format sind im Datentyp Display/Variable 163 gespeichert. Display bedeutet, daß die Datei aus lesbaren ASCII-Zeichen besteht.

TIPS & TRICKS

Variable 163 gibt die Anzahl der Bytes pro Datensatz an. Im Gegensatz zu Fixed-Dateien (alle Datensätze haben gleiche Länge), können diese unterschiedliche Länge haben (bis zu 163 Bytes!).
Output gibt nur an, daß etwas ausgegeben werden soll (hier speziell auf Diskette!).
Im Programm würde das ganze dann folgendermaßen aussehen:

```
10 OPEN#1:"DSK1.
TEST",OUTPUT,DIS-
PLAY,VARIABLE 163
```

Nachdem man die Datei eröffnet hat, muß man einen kleinen String definieren, der folgendermaßen aufgebaut ist:

I) Die Zeilennummer: Man Man benötigt zwei CHR§ für die Zeilennummer. Der erste CHR§ ist das High-Byte der Zeilennummer (High-Byte, d.h. es wird mit 256 multipliziert), der zweite CHR§ ist das Low-Byte (es wird nicht multipliziert!). Möchte man jetzt z.B. die Zeilennummer 31500 erhalten, so teilt man 31500 durch 256 – man erhält 123,04... (die Nachkommastellen einfach wegfallen lassen). 123 ist also das High-Byte). Zum Berechnen des Low-Bytes rechnet man: 31500–123*256 und erhält 12!

II) Das Token: Der dritte CHR§ gibt das Token an, daß man verwenden möchte (siehe Liste). Wir wählen für unseren Test ein "!" aus – dies hat den Wert 131. Der Token wird nur dazu verwendet, damit es im Programm eine normale REM-Zeile ergibt und keine Fehlermeldung auftritt.
Ihnen wird bestimmt aufgefallen sein, daß einige Token anscheinend fehlen, dies ist aber nicht der Fall. Manche CHR§-Werte haben kein Token zugeordnet.

III) Nun kommt nur noch eine Laufvariable, die den eigentlichen Token angibt – wir nehmen eine FOR...NEXT-Schleife von 129 bis 255. Warum erst ab 129? Ganz einfach, denn erst ab 129 stehen die definierten Charakter (123ABC...).
Das Programm, es stellt übrigens alle Token nochmals dar, sieht dann so aus:

```
20 FOR I=129 TO 255
30 A§=CHR§(0)&CHR§
   ((I-126)*10)&CHR§
   (131)&CHR§(I)&
   CHR§(0)&CHR§(0)
40 PRINT #1:A§
50 NEXT I
60 PRINT #1:CHR§(255)
   &CHR§(255)
70 CLOSE #1
80 END
```

In Zeile 60 wird die Print#1-Anweisung nur dazu verwendet, die Mergedatei, die im Programm generiert wird, zu schließen. Close#1 in Zeile 70 schließt nur die in Zeile 10 eröffnete Datei!

Das Programm generiert wiederum ein Programm, das nur aus "!" und nachgestellten Basic Token besteht – dieses kann man nun mit MERGE DSK1. TEST einlesen und listen.

Spezielle Anwendungen

Möchte man nun aus einem Programm (z.B. Zeichendefinitionsprogramm) die Charakter in Form von Data's abspeichern, geht man so vor:

```
10 OPEN#:"DSK1.
   DATA",OUTPUT,
   DISPLAY,VARIABLE
   163
20 FOR I=CHARTSTART
   TO CHAREND
30 CALL CHARPAT
   (I,B§)
40 A§=CHR§(0)&CHR§
   ((I-(CHARTSTART-1))
   *10)&CHR§(147)&
   CHR§(200)&CHR§(16)
   B§&CHR§(0)&CHR§
   (0)
50 PRINT#1:A§
60 NEXT I
70 PRINT #1:CHR§(255)
   &CHR§(255)
80 CLOSE #1
90 END
```

Erläuterungen:

CHARSTART gibt den ersten Charaktercode, CHAREND den letzten Code, der in Data's geschrieben werden soll, an! In A§ geben die ersten beiden CHR§ nochmals die Zeilennummer an (in Zehnerschritten!). Das dritte CHR§ gibt das Token an (hier: Data). CHR§(200) bzw. CHR§ (16) müssen nach dem Data-Token stehen, da sonst ein Error auftritt. Diese beiden Zahlen geben den internen Code an. Man könnte sie auch weglassen, nur müßte man in jeder Data-Zeile am Ende ein/zweimal Delete drücken... und das möchte ganz bestimmt niemand!
Weiterhin gibt es viele Möglichkeiten wie man die Token benutzen und miteinander verbinden kann. Für andere Computer gibt es ein Programm, das selbst ein Programm aufbaut.
Der Phantasie sind hier fast keine Grenzen gesetzt. Zum Schluß noch einen kleinen Gag:
Normalerweise gibt es die Zeile 0 nicht. Man kann trotzdem, zum Staunen und Verzweifeln der Freunde, eine „nullte" Zeile über diese Methode bekommen...

1) Datei wie oben eröffnen
2) einen String definieren, der in der Zeile stehen soll, z.B. c§=" (C) 1985 Oliver Siffrin"
3) den zweiten String wie folgt aufbauen: A§=CHR§(0)&CHR§(0)&CHR§(131)&C§& CHR§(0)&CHR§(0)
4) die restliche Prozedur wie bei den anderen Beispielen...

Oliver Siffrin

Anmerkung: § = $

```
90 REM BEISPIEL 1 ZU CHR$-KU
NDE
100 OPEN #1:"DSK1.TEST",OUTP
UT,DISPLAY ,VARIABLE 163
110 FOR I=129 TO 255
120 A$=CHR$(0)&CHR$((I-126)*
10)&CHR$(131)&CHR$(I)&CHR$(0
)&CHR$(0)
130 PRINT #1:A$
140 NEXT I
150 PRINT #1:CHR$(255)&CHR$(
255)
160 CLOSE #1
170 END
```

```
90 REM BEISPIEL 2 ZU CHR$-KU
NDE
100 OPEN #1:"DSK1.TEST",OUTP
UT,DISPLAY ,VARIABLE 163
110 FOR I=CHARSTART TO CHARE
ND
120 CALL CHARPAT(I,B$)
130 A$=CHR$(0)&CHR$((I-CHARS
TART-1)*10)&CHR$(147)&CHR$(2
00)&CHR$(16)&B$&CHR$(0)&CHR$
(0)
140 PRINT #1:A$
150 NEXT I
160 PRINT #1:CHR$(255)&CHR$(
255)
170 CLOSE #1
180 END
```

Herzlichen Glückwunsch zum neuen PC!

Die Würfel sind gefallen: Fred Tiedemann aus Hamburg 11 kann sich freuen. Aus mehr als 11000 Einsendungen zu unserem großen Gewinnspiel zog unsere Glücksfee seinen Namen. Er gewann den großen Preis: Einen PC 10 von Commodore im Wert von rund 5500 DM.



Hauptgewinn: Ein PC 10 von Commodore!

Herzlichen Glückwunsch von der Isar an die Alster. Weitere zwei Leser werden noch eine Weihnachtsüberraschung erleben: Sie erhalten je einen MSX-Computer Yashica 64 - pünktlich zum Weihnachtsfest. Es sind: Eckhard Spaeth aus 3325 Lengede und Henning Neumann, 5040 Leverkusen.

Ihnen allen wünscht die Redaktion viel Freude und Spaß.

Convert erspart das Abtippen von List-Dateien

Stellen Sie sich vor, Sie bekommen z.B. durch Ihre RS232 den List(ASCII)-Datei eines Basic-Programmes geschickt. Sobald dieses Programm länger als 20 Zeilen ist, wird es mühsam, es abzutippen, denn der TI 99/4A erkennt ja ein derartiges Format nicht als Programm. Die Fehler rechnet man damit am besten nicht ein.

Dies zu vermeiden hilft das Programm CONVERT, welches als EPROM für die in der TI-REVUE 5/85 vorgestellte Modulplatine geliefert wird. Für Insider: Es ist in GPL geschrieben. Es wandelt TI-Basic oder Ex-Basic List-Files wieder in das Program-Format um, so daß der TI 99 sie als Programm verarbeiten kann. D.h. ein List-File, der im Variable 80 Format auf Diskette abgespeichert ist, wird in Basic-Programm umgewandelt, welches geladen und ausgeführt werden kann. Somit fällt jegliches Abtippen von List-Files, welche auf Diskette abgespeichert sind, weg.
CONVERT ist ein Programm, welches auch ohne Speichererweiterung, jedoch sinnvollerweise mit mindestens einem Diskettenlaufwerk, lauffähig ist. Es funktioniert aber auch der Empfang der Listfiles über die RS232-Schnittstelle und die Speicherung des fertigen Programms auf Kassette.
Beim Anwählen von CONVERT erscheint eine Hauptwahlliste, welche abfragt, welche Basic-Version umgewandelt werden soll, wobei hier eine besondere Funktion namens M-Basic die in anderen Basicdialekten häufig verwendeten ?, MIDS und das einfache Trennungszeichen korrekt zu allen X-Basic-Befehlen für den TI 99 umwandelt. Es ist also von Nutzen, sich zu vergewissern, in welchem Basic-Dialekt das Programm, das im List-File enthalten ist, geschrieben wurde. Nach der Wahl des Dialektes muß man noch den Namen des List-Files eingeben, mit welchem er auf der Diskette abgespeichert ist. Ist dies geschehen, so zeigt der Compu-

UNSER ☎ TELEFONSERVICE ☎

Leserbriefe und Fragen sind uns stets willkommen. Wir beantworten sie entweder direkt oder auf der Leserbriefseite

ACHTUNG !!!

Wie immer steht unseren Lesern unser Telefon-Service zur Verfügung! Jeden Dienstag von 15 bis 19 Uhr. Für technische Fragen: 0731/33 220 und für Listings/Programme: 089/129 80 13

ter auf dem Bildschirm die Zeilennummer an, welche gerade umgewandelt wird. Danach gibt man dann den Namen ein, mit dem das neu entstandene Programm auf der Diskette oder Kassette abgelegt werden soll, so speichert der Computer es ab, und das neu entstandene Programm kann nun normal als Basicprogramm geladen werden.
Tritt während der Umwandlung ein Fehler auf, so wird diese Zeile, in der er auftrat, nicht umgewandelt, sondern es erscheint die entsprechende Zeile als REM Zeile, in der vermerkt steht, daß eine Umwandlung für das Programm unmöglich war.
Einen Haken hat die ganze Sache jedoch: Basic-Zeilen können bis zu 254 Zeichen lang sein, die List-Dateien sind aber nur bis max. 80 Zeichen lang. Wird dadurch ein Befehl, ein Variablenname oder auch ein String in Anführungszeichen in zwei Hälften geteilt, so ist das Programm nicht in der Lage, diesen umzuwandeln. Es tritt dann ein SYNTAX ERROR auf. Es wandelt diesen Befehl erst dann um, wenn man den betreffenden Befehl in eine neue Zeile gesetzt hat, denn das Modul erlaubt schon längere Zeilen als 80 Zeichen, nur darf dann am Beginn der neuen Zeile keine Zahl stehen. Dies ist möglich, da man einen List-File anhand der Editor/Assembler Moduls oder des TI Writers editieren und dadurch verändern kann. Hat man keines der beiden Module, so muß man entweder diese Zeile abtippen, oder sich ein eigenes Editierprogramm schreiben, welches sehr gut in Basic zu bewerkstelligen ist. Trotzdem ist das CONVERT-Programm eine große Arbeitserleichterung, welche vor allem bei den DFÜ-User begrüßt werden dürfte.

Oliver Huber

ZEHNERBLOCK

Dieses Unterprogramm soll dem Anwender häufige Zahleneingaben erleichtern.
Der Aufruf des Unterprogrammes erfolgt durch CALL BLOCK(ZAHL, ZEILE, SPALTE) und funktioniert ähnlich wie ACCEPT AT.
Die Alpha-Lock-Taste muß für die Zahleneingabe gelöst sein!
Die Tastaturbelegung ist folgende:

7(7) 8(8) 9(9) 0(0) /(–)
u(4) i(5) o(6)
j(1) k(2) l(3) .(.)

Das Minuszeichen kann ohne SHIFT eingegeben werden.
Falscheingaben sind nicht möglich.
Das zugehörige Testprogramm ist ein einfaches Additionsprogramm (Anzeige der Summe durch Eingabe von 0 oder " ") und soll nur die Funktionsfähigkeit des Unterprogramms demonstrieren.

Rudolf Hoven

```
1 !*************************
2 !*      Zehner-Block      *
3 !*       TI/99/4A         *
4 !*       EXT.BASIC        *
5 !*     Rudolf Hoven       *
9 !*************************
10 !
20 !      Speicherbelegung
30 !      des Unterprogramms:
40 !          604 Bytes
50 !
90 REM *** TEST-ANFANG ***
100 DISPLAY AT(2,3)ERASE ALL
:"ALPHA-LOCK LOESEN ! ! !"
110 DISPLAY AT(12,5):"DATA?"
120 CALL BLOCK(ZAHL,12,11)
130 IF ZAHL THEN SUM=SUM+ZAH
L :: GOTO 120
140 DISPLAY AT(12,5):"SUMME=
";SUM
200 CALL KEY(0,K,S):: IF S=0
 THEN 200 ELSE SUM=0 :: GOTO
 110
210 REM *** TEST-ENDE ***
220 !
230 !
240 !
250 !
30000 SUB BLOCK(ZAHL,ZEI,SPA
)
30020 ON ERROR 30360
30040 IF F THEN 30120
30080 F=1 :: RESTORE 30380 :
: FOR I=1 TO 7 :: READ QW,TY
,ES(I),PS(I):: CALL CHARPAT(
QW,C$):: CALL CHAR(TY,C$)::
NEXT I
30120 ACCEPT AT(ZEI,SPA)BEEP
 SIZE(-13)VALIDATE("jkluio78
90./"):X$
30160 IF X$="" THEN ZAHL=0 :
: SUBEXIT
30180 IF SEG$(X$,1,1)="/" TH
EN X$="-"&SEG$(X$,2,LEN(X$)-
1)
30200 FOR J=1 TO LEN(X$):: P
,E=ASC(SEG$(X$,J,1))
30220 FOR I=1 TO 6
30240 IF E=ES(I)THEN P=PS(I)
:: GOTO 30280
30260 NEXT I
30280 YOK$=YOK$&CHR$(P):: NE
XT J
30320 ZAHL=VAL(YOK$):: YOK$=
"" :: SUBEXIT
30360 CALL SOUND(250,-3,0)::
 ON ERROR 30360 :: YOK$="" :
: RETURN 30120
30380 DATA 49,106,106,49,45,
47,117,52,50,107,105,53,51,1
08,111,54,52,117,107,50,53,1
05,108,51,54,111,0,0
30420 SUBEND
```

KALENDER

Dieses Programm bietet Ihnen zwei Funktionen zur Auswahl. Sie können sich den Wochentag zu einem beliebigen Datum (nach 1582 , zu diesem Zeitpunkt wurde der Kalender umgestellt) errechnen lassen oder mit Funktion 2 einen Jahreskalender für ein beliebiges Jahr, nach Ihrer Wahl, ausdrukken lassen.
In diesem Programm wurde ein Binder 8510A benutzt. Es läßt sich aber auch leicht für jeden anderen Drucker umschreiben. Der Computer überprüft die eingegebenen Daten und weist solche, die nicht existieren, zurück.
Das Programm ist durch geschickte Programmierung sehr schnell und genau und durch die guten Dokumentationen im Listing leicht nachvollziehbar. *H.-J. Adler*

APARAK 5

Aparak 5 ist ein Kurvendiskussions-Programm, welches Parametergleichungen zeichnet und/oder ihre Nullstellen und Extremwerte berechnet. Zum Unterschied zu vielen anderen ähnlichen Programmen kann hier die Funktion direkt während des Ablaufes eingegeben werden und wird dann in die entsprechenden Zeilen gepoked, in denen zu Beginn eine Beispielfunktion eingegeben ist.
Bei der Eingabe von Aparak 5 ist besonders auf die korrekte Eingabe der REM-Zeilen zu achten. Eine Änderung von ihnen oder eine Änderung der Zeilennummern kann zu einer Fehlfunktion des gesamten Programmes führen. Nach der Eingabe des Programmes und nach dem Starten mit RUN erscheint ein erstes Menü auf dem Bildschirm. Bei diesem kann man entweder die Gleichungen im Intervall von -5 bis 5 auf dem Bildschirm darstellen lassen oder das Hauptmenü anwählen. Im Hauptmenü werden alle Werte, die man verändern kann, aufgeführt. Außer der Funktion. Diese wird nur bei der Wahl von Punkt 9 des Hauptmenüs gezeigt. Die Eingabe des Intervalls erfolgt durch Trennung mit der ENTER-Taste der zwei Werte. Der mittlere X oder Y-Wert gibt an, welche Koordinate des Achsenkreuzes in die Mitte des Zeichenbildschirmes gelegt wird. Der Maßstab gibt den Kehrwert des Faktors an, mit dem die ganze Funktion vergrößert oder verkleinert dargestellt wird. Der Punkt 8 des Hauptmenüs zeigt an, ob die einzelnen Punkte der Funktion beim Zeichnen mit einer Linie verbunden werden sollen oder nicht. Das Programm ist für den TI 99/4A + 32 K-Bytes + Apesoft-Grafik oder TI 99/4A + 32 K-Bytes + X-BASIC 2 Plus angelegt.

Anmerkung der Redaktion:
Obwohl in der Dokumentation des Listings schon darauf aufmerksam gemacht wurde, sei an dieser Stelle noch einmal ausdrücklich davor gewarnt, irgend etwas an dem Listing zu ändern.
Die entsprechenden Funktionen müssen nicht, wie im X-Basic, in DATA-Zeilen eingegeben werden, sondern sie werden in die entsprechenden Zeilen gepoked. Wird an dem Listing irgend etwas geändert, kann es zu Fehlfunktionen kommen, deshalb haben wir auch auf unser bekanntes Kopfbild verzichtet.
Also noch einmal: Keine REM-Zeilen löschen, keine Zeilen anfügen oder weglassen und die Zeilen nicht umnumerieren!!!
Da es für die Ape-Soft-Grafik oder das Extended Basic II Plus bisher kaum Programme gibt (mit Ausnahme der Programme von Dr. Alma Peschitz in TI 99 Special I und II), werden sich Besitzer dieser Software sicherlich über Zuwachs ihrer Programmsammlung freuen.
Im Gegensatz zu Kurvendiskussionen im normalen X-Basic, bei denen man normalerweise viel Zeit und Geduld aufbringen muß, wird hier mit Hilfe der Assembler-Unterroutinen die Funktion in Sekundenschnelle auf dem Bildschirm generiert. Das Menü bietet mannigfaltige Optionen an.
Jeder TI-User, der über die Ape-Soft-Grafik oder das Extended Basic II Plus verfügt, und die erforderlichen mathematischen Kenntnisse verfügt, wird von diesem Programm begeistert sein.

BAUSTATIK

```
DURCHBIEGUNG  IN  FELDMITTE
=============================================

EINFELDTRAEGER LAGERUNG:GELENKIG ODER EINGESPANNT.

TRAEGHEITSMOMENT IN CM^4= 11770
SPANNWEITE IN METER= 7.7

LASTEN SIEHE STATIKSKIZZE
-----------------------------------------
Y (O) IN FELDMITTE =      .59 CM
Y (F)       -//-   =      1.46 CM
-----------------------------------------
SUMME Y            =      2.05 CM
=========================================

DURCHBIEGUNG ENTSPRICHT  1: 375.61 DER SPANNWEITE.
---------------------------------------------
```

Dieses Programm ist für Baustatiker, die ein Programm zur Schnittkraftermittlung besitzen, eine nützliche Ergänzung, da die Durchbiegung in manchen statischen Berechnungen das Bemessungskriterium ist.
Das Programm arbeitet im Dialog mit Ihnen, daher sind keine besonderen Anleitungen erforderlich.
Folgende Belastungen sind möglich:
1) Gleichlasten
2) Bel. viele Einzellasten
3) Teilstreckenlasten werden durch beliebig viele Einzellasten definiert.

Auf Wunsch werden die Ergebnisse über einen Drucker ausgegeben.
J.Picksak

```
100 ! *********************
110 ! *                   *
120 ! *     Kalender      *
130 ! *   Version  2.1    *
140 ! *   Copyright by    *
150 ! *  Hans-j.  Adler   *
160 ! * Eagle  Computing  *
170 ! *                   *
180 ! *  Bearbeitet vom   *
190 ! *     Team des      *
200 ! * Aktuell  Verlages *
210 ! *     Muenchen      *
220 ! *                   *
230 ! *Benoetigte Geraete:*
240 ! *  TI99/4A  Konsole *
250 ! *     Ext. Basic    *
260 ! *Disk  od. Cass.Rec.*
270 ! *      Drucker      *
280 ! * Speicherbelegung: *
290 ! *    7178  Bytes    *
300 ! *                   *
310 ! *********************
320 !
330 ! Vorbereitungen
340 ! ==============
350 CALL CLEAR :: ON BREAK N
EXT
360 DIM TA(12),MO(12),DA(12)
,D$(7)
370 DEF F=Y*365+D+31*(M-1)
380 DATA 31,28,31,30,31,30,3
1,31,30,31,30,31
390 GOTO 420 :: CALL CHAR ::
 CALL SPRITE :: CALL DELSPRI
TE :: CALL MAGNIFY
400 CALL CHARPAT :: CALL SOU
ND :: CALL POSITION :: CALL
MOTION :: CALL LOCATE
410 F1,C$,I,J,X,Y,A$,K,S,Y1,
D1,M1,D,M,Y,FL,SCH,B,BB,JA$,
A1$,B$,MON,P :: !@P-
420 DATA Sonntag,Montag,Dien
stag,Mittwoch,Donnerstag,Fre
itag,Sonnabend
430 DATA 75,65,76,69,78,68,6
9,82
440 DATA 60,66,153,165,165,1
29,66,60,0,0,0,127,73,73,73,
65,96,144,144,0,0,0,0
450 C$=CHR$(27)
460 !  Daten einlesen
470 FOR I=1 TO 12 :: READ MO
(I):: NEXT I
480 FOR I=1 TO 7 :: READ D$(
I):: NEXT I
490 CALL CHAR(64,"3C4299A1A1
99423C",93,"0044004444444438
",95,"00FFFF")
500 CALL CHAR(123,"000044003
8447C44",124,"000044007C4444
7C0000440044444438")
510 !
520 !  Titelbild
530 !  =========
540 CALL COL(16,2,5):: CALL
MAGNIFY(2)
550 FOR I=1 TO 4 :: READ J
560 CALL SPRITE(#I,J,10,240,
16+I*24,25,0)
570 CALL POSITION(#I,X,Y)::
IF X<51+I*4 OR X>61+I*4 THEN
 570
580 CALL MOTION(#I,0,0):: CA
LL LOCATE(#I,56+I*4,16+I*24)
590 NEXT I
600 FOR I=5 TO 8 :: READ J
610 CALL SPRITE(#I,J,10,240,
16+I*24,-25,0)
620 CALL POSITION(#I,X,Y)::
IF X>62+I*4 OR X<51+I*4 THEN
 620
630 CALL MOTION(#I,0,0):: CA
LL LOCATE(#I,56+I*4,16+I*24)
640 NEXT I
650 FOR J=1 TO 3 :: FOR I=20
0 TO 400 STEP 50 :: CALL SOU
ND(70,I,I/100):: NEXT I
660 NEXT J :: FOR I=400 TO 2
00 STEP -40 :: CALL SOUND(70
,I,I/100):: NEXT I
670 A$="@ EAGLE-computing '8
4......."
680 DISPLAY AT(23,1):A$
690 A$=SEG$(A$,28,1)&A$ :: A
$=SEG$(A$,1,28)
700 CALL KEY(0,K,S):: IF S=0
 THEN 680 ELSE CALL DELSPRIT
E(ALL)
710 !
720 !  Menue
730 !  =====
740 CALL COL(2,8,14)
750 CALL SPRITE(#1,77,2,15,8
9,#2,69,2,15,113,#3,78,2,15,
137,#4,93,2,15,161)
760 DISPLAY AT(5,1):RPT$("_"
,28)
770 DISPLAY AT(13,3):"Wochen
tag bestimmen...." :: CALL S
PRITE(#5,49,2,89,217)
780 DISPLAY AT(16,3):"Kalend
er drucken......." :: CALL S
PRITE(#6,50,2,113,217)
790 DISPLAY AT(19,3):"Ende..
................." :: CALL S
PRITE(#7,51,2,137,217)
800 CALL KI(49,51,K):: CALL
DELSPRITE(ALL)
810 ON K GOTO 850,1250,2040
820 !
```

```
830 !  Wochentag bestimmen
840 !  =================
850 CALL COL(2,16,4)
860 DISPLAY AT(2,5):"WOCHENT
AG BESTIMMEN":RPT$("_",28)
870 IF Y1>Ø THEN 94Ø
88Ø DISPLAY AT(6,1):"Heutige
s Datum : Ø1.Ø1.1984"
89Ø ACCEPT AT(6,18)SIZE(-2)V
ALIDATE(DIGIT):D1 :: IF D1=Ø
 OR D1>31 THEN 89Ø
9ØØ ACCEPT AT(6,21)SIZE(-2)V
ALIDATE(DIGIT):M1 :: IF M1=Ø
 OR M1>12 THEN 9ØØ
91Ø CALL DATE(D1,M1,FL,MO())
:: IF FL=1 THEN FL=Ø :: GOTO
 89Ø
92Ø ACCEPT AT(6,26)SIZE(-2)V
ALIDATE(DIGIT):Y1 :: IF Y1<8
4 THEN 92Ø
93Ø Y1=19ØØ+Y1
94Ø DISPLAY AT(9,1):"Gefragt
es Datum: ØØ.ØØ.ØØØØ"
95Ø ACCEPT AT(9,18)SIZE(-2)V
ALIDATE(DIGIT):D :: IF D=Ø O
R D>31 THEN 95Ø
96Ø ACCEPT AT(9,21)SIZE(-2)V
ALIDATE(DIGIT):M :: IF M=Ø O
R M>12 THEN 96Ø
97Ø CALL DATE(D,M,FL,MO())::
 IF FL=1 THEN FL=Ø :: GOTO 9
5Ø
98Ø ACCEPT AT(9,24)SIZE(-5)V
ALIDATE(DIGIT):Y
99Ø IF Y>=1582 THEN 1Ø3Ø
1ØØØ DISPLAY AT(17,1):"Es k|
nnen keine Wochentage": :"vo
r 1582 angegeben werden !!"
1Ø1Ø FOR I=1 TO 1ØØØ :: NEXT
 I
1Ø2Ø DISPLAY AT(17,1):"": :"
" :: GOTO 95Ø
1Ø3Ø IF D=29 AND M=2 THEN SC
H=1 :: D=1 :: M=3
1Ø4Ø GOSUB 215Ø :: IF SCH=Ø
THEN 111Ø
1Ø5Ø !
1Ø6Ø !  Existiert 29.2 ?
1Ø7Ø BB=B :: D=29 :: M=2 ::
GOSUB 215Ø :: SCH=Ø
1Ø8Ø IF BB<>B THEN 111Ø
1Ø9Ø DISPLAY AT(17,1):"Der 2
9.Ø2."&STR$(Y)&" existiert":
 :"nicht !!!"
11ØØ GOTO 12ØØ
111Ø A$="Der "&STR$(D)&"."&S
TR$(M)&"."&STR$(Y)
112Ø IF (Y1*12+M1)*31+D1<(Y*
12+M)*31+D THEN 116Ø
113Ø IF (Y1*12+M1)*31+D1=(Y*
12+M)*31+D THEN 115Ø
114Ø DISPLAY AT(17,1):A$&" w
ar ein" :: GOTO 117Ø
115Ø DISPLAY AT(17,1):A$&" i
st ein" :: GOTO 117Ø
116Ø DISPLAY AT(17,1):A$&" w
ird ein" :: X=1
117Ø DISPLAY AT(19,1):D$(B)
118Ø IF X=1 THEN X=Ø :: DISP
LAY AT(19,LEN(D$(B))+2):"sei
n." :: GOTO 12ØØ
119Ø DISPLAY AT(19,LEN(D$(B)
)+1):"."
12ØØ DISPLAY AT(23,27):">>"
121Ø CALL KI(13,127,K):: GOT
O 74Ø
122Ø !
123Ø !  Kalender drucken
124Ø !  ================
125Ø CALL COL(2,4,7)
126Ø DISPLAY AT(2,7):"KALEND
ER DRUCKEN":RPT$("_",28)
127Ø DISPLAY AT(12,1):"Welch
es Kalenderjahr ? ØØØØ"
128Ø ACCEPT AT(12,24)SIZE(-5
)VALIDATE(DIGIT):Y :: IF Y<1
582 THEN 128Ø
129Ø !  12 Monatserste berec
hnen - D(1)...D(12)
13ØØ !  ------------------
-------------------
131Ø D=1
132Ø FOR I=1 TO 12 :: M=I
133Ø GOSUB 215Ø
134Ø DA(I)=B-1
135Ø NEXT I
136Ø !  Schaltjahr-Check
137Ø !  ---------------
138Ø D=29 :: M=2 :: GOSUB 21
5Ø
139Ø IF DA(3)=B THEN MO(2)=2
9 ELSE MO(2)=28 :: GOTO 141Ø
14ØØ DISPLAY AT(16,2):STR$(Y
);" ist ein Schaltjahr !"
141Ø DISPLAY AT(21,4):"Druck
er einschalten -": :".....Ta
ste dr}cken ..."
142Ø CALL KI(13,127,K):: DIS
PLAY AT(21,1):"": :"...Druck
vorgang l{uft ..."
143Ø OPEN #1:"RS232.BA=96ØØ.
DA=8.TW"
144Ø PRINT #1:C$&"LØØ3";C$&"
>";C$&"T16": :
145Ø !  Jahreszahl drucken
146Ø !  ------------------
147Ø JA$=STR$(Y)
148Ø FOR I=2 TO 8 :: PRINT #
1:TAB(4Ø-LEN(JA$)*8);C$&"!";
CHR$(14);
```

```
1490 FOR J=1 TO LEN(JA$):: A
$=SEG$(JA$,J,1):: CALL CHARP
AT(ASC(A$),A1$)
1500 A1$=SEG$(A1$,2*I-1,2)
1510 FOR K=1 TO 2 :: B$=SEG$
(A1$,K,1)
1520 IF B$="0" THEN PRINT #1
:"....";:: GOTO 1680
1530 IF B$="1" THEN PRINT #1
:"...*";:: GOTO 1680
1540 IF B$="2" THEN PRINT #1
:"..*.";:: GOTO 1680
1550 IF B$="3" THEN PRINT #1
:"..**";:: GOTO 1680
1560 IF B$="4" THEN PRINT #1
:".*..";:: GOTO 1680
1570 IF B$="5" THEN PRINT #1
:" * *";:: GOTO 1680
1580 IF B$="6" THEN PRINT #1
:" ** ";:: GOTO 1680
1590 IF B$="7" THEN PRINT #1
:" ***";:: GOTO 1680
1600 IF B$="8" THEN PRINT #1
:"*...";:: GOTO 1680
1610 IF B$="9" THEN PRINT #1
:"*..*";:: GOTO 1680
1620 IF B$="A" THEN PRINT #1
:"* * ";:: GOTO 1680
1630 IF B$="B" THEN PRINT #1
:"* **";:: GOTO 1680
1640 IF B$="C" THEN PRINT #1
:"**..";:: GOTO 1680
1650 IF B$="D" THEN PRINT #1
:"** *";:: GOTO 1680
1660 IF B$="E" THEN PRINT #1
:"*** ";:: GOTO 1680
1670 IF B$="F" THEN PRINT #1
:"****";
1680 NEXT K :: NEXT J :: PRI
NT #1:" ";CHR$(15)
1690 NEXT I :: PRINT #1:C$&C
HR$(34);C$&"A": :
1700 !  Kalender
1710 !  --------
1720 FOR I=1 TO 4 :: PRINT #
1:RPT$("*",76):: GOSUB 2120
1730 ON I GOTO 1740,1760,178
0,1800
1740 PRINT #1:"*.........JAN
UAR.........*........FEBRUAR
.........*..........MIRZ....
......*"
1750 GOSUB 2120 :: PRINT #1:
RPT$("*",76):: GOTO 1820
1760 PRINT #1:"*.........APR
IL..........*..........MAI..
.........*..........JUNI....
......*"
1770 GOSUB 2120 :: PRINT #1:
RPT$("*",76):: GOTO 1820
1780 PRINT #1:"*..........JU
LI.........*.........AUGUST
.........*.......SEPTEMBER..
......*"
1790 GOSUB 2120 :: PRINT #1:
RPT$("*",76):: GOTO 1820
1800 PRINT #1:"*.........OKTO
BER.........*........NOVEMBE
R........*........DEZEMBER..
......*"
1810 GOSUB 2120 :: PRINT #1:
RPT$("*",76)
1820 PRINT #1:"*..So Mo Di M
i Do Fr Sa..*..So Mo Di Mi D
o Fr Sa..*..So Mo Di Mi Do F
r Sa..*" :: GOSUB 2120
1830 MON=3*(I-1)+1
1840 FOR J=1 TO 6 :: FOR K=M
ON TO MON+2
1850 PRINT #1:"*..";:: IF J<
>1 THEN 1900
1860 IF DA(K)=0 THEN 1880
1870 FOR P=1 TO DA(K):: PRIN
T #1:"...";:: NEXT P
1880 FOR P=1 TO 7-DA(K):: PR
INT #1,USING "## ":P;:: NEXT
 P :: PRINT #1:" ";
1890 TA(K)=7-DA(K):: GOTO 19
30
1900 FOR P=1 TO 7 :: IF TA(K
)>=MO(K)THEN PRINT #1:"   ";
:: GOTO 1920
1910 TA(K)=TA(K)+1 :: PRINT
#1,USING "## ":TA(K);
1920 NEXT P :: PRINT #1:" ";
1930 NEXT K :: PRINT #1:"*";
1940 PRINT #1:" " :: NEXT J
1950 GOSUB 2120 :: NEXT I ::
 PRINT #1:RPT$("*",76)
1960 RESTORE 440 :: PRINT #1
: :CHR$(14);C$&"S0022";
1970 FOR I=1 TO 22 :: READ J
 :: PRINT #1:CHR$(J);:: NEXT
 I
1980 PRINT #1:CHR$(15);"'84"
;
1990 PRINT #1:C$&"<";C$&"L00
0";CHR$(13);CHR$(12):: CLOSE
 #1
2000 GOTO 740
2010 !
2020 !  Ende
2030 !  ----
2040 CALL CLEAR :: STOP
2050 !
2060 !
2070 !  SUBROUTINES
2080 !  ***********
2090 !
```

```
2100 !
2110 !  Printer-Routine
2120 PRINT #1:"*"&RPT$(" ",2
4)&"*"&RPT$(" ",24)&"*"&RPT$
(" ",24)&"*" :: RETURN
2130 !
2140 !  Tag berechnen
2150 IF M>2 THEN 2170
2160 F1=F+INT((Y-1)/4)-INT(.
75*(INT((Y-1)/100)+1)):: GOT
O 2180
2170 F1=F-INT(.4*M+2.3)+INT(
Y/4)-INT(.75*(INT(Y/100)+1))
2180 B=F1+INT(-F1/7)*7
2190 B=B-(B<=0)*7+(B>6)*7
2200 RETURN
2210 !@P+
2220 !
2230 !  Farben definieren
2240 SUB COL(X,Y,Z)
2250 CALL CLEAR :: CALL SCRE
EN(Z)
2260 FOR I=0 TO 14 :: CALL C
OLOR(I,X,Y):: NEXT I
2270 SUBEND
2280 !
2290 !  Call Key
2300 SUB KI(X,Y,K)
2310 CALL KEY(0,K,S):: IF K<
X OR K>Y THEN 2310 ELSE K=K-
48
2320 SUBEND
2330 !
2340 !  Datums-Check
2350 SUB DATE(X,Y,FL,MO())
2360 IF X>MO(Y)-(Y=2)THEN FL
=1
2370 SUBEND
```

BÖRSE


```
100 !ACHTUNG: JEDES WEITERE
     RES ODER JEDE WEITERE
     ZEILENNUMMERNAENDERUNG
     FUEHRT ZU EINER FEHL-
     FUNKTION DES PROGRAMMES
110 !
120 ! *********************
130 !**********************
140 !*                    *
     *                    *
     *    A P A R A K     *
     *                    *
150 !*   (c) by O. Huber  *
160 !*                    *
170 !*  Pfeifenmacherg. 6 *
180 !*                    *
190 !* 7900 Ulm-Soeflingen *
200 !*                    *
     *         &          *
     *                    *
220 !*  (c) Heiko Liebald *
     *                    *
     *     Eichenweg 7    *
     *                    *
230 !*  7914 Pfaffenhofen *
     *                    *
     *                    *
     **********************
240 ! *********************
250 !
260 ! Programmed: 1985
270 !
280 X$="D" :: Y$="4*SIN(D)"
:: GOTO 670
290 GOTO 320
300 DEF X(D)=D :: RETURN ::
!AAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAA
310 DEF Y(D)=4*SIN(D):: RETU
RN :: !AAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAA
320 REM APARAK2 & ZEILENPOKE
330 ZZ=300
340 CALL SCREEN(2)
350 CALL PEEK(-31950,A,B)::
EE=A*256+B-65537
360 CALL PEEK(-31952,A,B)::
SS=A*256+B-65536
370 CALL PEEK(EE-2,A,B):: IF
 A*256+B<>ZZ THEN EE=EE-4 EL
SE 390
380 IF EE>SS THEN 370 ELSE P
RINT "ZEILE NICHT DA" :: END
390 CALL PEEK(EE,A,B):: CC=A
*256+B-65536
400 PRINT CC :: CALL PEEK(CC
```

```
,R):: IF R<>131 AND R<>154 A
ND R<>137 THEN PRINT "KEIN '
REM'" :: END
410 IF ZZ=300 THEN CALL LOAD
(CC,137,88,183,68,182,190)::
 CC=CC+6 ELSE CALL LOAD(CC,1
37,89,183,68,182,190):: CC=C
C+6
420 IF ZZ=300 THEN INPUT "X(
D)=":X$ :: FX$=X$ ELSE INPUT
 "Y(D)=":Y$ :: FX$=Y$
430 DATA ABS,ATN,COS,EXP,INT
,LOG,SGN,SIN,SQR,TAN
440 PP=1
450 FOR AA=1 TO LEN(FX$)
460 B=ASC(SEG$(FX$,AA,1))
470 IF B=43 THEN DD=193 ELSE
 IF B=45 THEN DD=194 ELSE IF
 B=42 THEN DD=195 ELSE IF B=
47 THEN DD=196
480 IF B=40 THEN DD=183 ELSE
 IF B=41 THEN DD=182
490 IF B=94 THEN DD=197
500 IF "PI"=SEG$(FX$,AA,2)TH
EN DD=221 :: AA=AA+1 :: GOTO
 620
510 RESTORE 430
520 FOR M=1 TO 10
530 READ B$
540 IF B$=SEG$(FX$,AA,3)THEN
 CALL LOAD(CC+PP-1,M+202)::
AA=AA+2 :: GOTO 640
550 NEXT M :: Q=1
560 IF B<>46 AND B<48 OR B>5
7 THEN 610
570 IF (B<>43 OR B<>45)AND A
A>2-Q THEN IF SEG$(FX$,AA+Q-
2,1)="E" THEN 590
580 IF B<>46 AND B<48 OR B>5
7 AND B<>69 THEN 590 ELSE Q=
Q+1 :: IF LEN(FX$)<Q+AA-1 TH
EN 590 ELSE B=ASC(SEG$(FX$,A
A+Q-1,1)):: GOTO 570
590 Q=Q-1 :: E$=SEG$(FX$,AA,
Q):: CALL LOAD(CC+PP-1,200,Q
):: FOR AAA=1 TO Q :: CALL L
OAD(CC+PP+AAA,ASC(SEG$(E$,AA
A,1))):: NEXT AAA
600 CC=CC+AAA :: AA=AA+Q-1 :
: GOTO 640
610 IF B>64 AND B<91 THEN DD
=B
620 IF DD=0 THEN PRINT "EING
ABEFEHLER" :: END
630 CALL LOAD(CC+PP-1,DD)
640 PP=PP+1 :: NEXT AA
650 CALL LOAD(CC+PP-1,130,13
6,130,131)
660 IF ZZ=300 AND HIL=1 THEN
 ZZ=310 :: GOTO 360
670 GOSUB 300 :: GOSUB 310
680 ON WARNING NEXT :: ON ER
ROR 1320
690 REM (C)BY OLI
        MAI / JUNI  '85
700 CALL CLEAR
710 CALL SCREEN(2)
720 A$="EIN" :: LI$="EIN"
730 FOR I=0 TO 14 :: CALL CO
LOR(I,4,1):: NEXT I
740 PRINT "U = ANHALTEN DES
ZEICHNENS": : :"C = NACH ANH
ALTEN WEITER": :"MALEN(NUR N
ACH ANHALTEN": :"MOEGLICH,NI
CHT NACH"
750 PRINT :"BEENDIGUNG DES Z
EICHNENS)": : :"0 = EXIT, NA
CHDEM MAN DAS": :"MENUE ANGE
WAEHLT HAT.": : : :
760 PRINT "1 = GEGEND UM NUL
L ZEICHNEN": :"2 = EIGENE DA
TEN"
770 CALL KEY(0,T,ST):: IF ST
=0 THEN 770
780 DA=-5 :: DE=5 :: MA=1
790 S=1 :: DS=1
800 IF T=49 THEN 820
810 IF T<>50 THEN 770 ELSE 1
010
820 !
830 CALL LINK("GRAFIC",0)
840 CALL LINK("WINDOW",6,10)
850 REM BILD ZEIGEN ******
860 IF A$<>"EIN" THEN 880
870 CALL LINK("AXIS",(63-XM*
10/MA),(64+XM*10/MA),(70-XM*
10/MA),(10/MA),(60-YM*10/MA)
,(60+YM*10/MA),(70-YM*10/MA)
,(10/MA))
880 CALL LINK("CENTRE",64-XM
*10/MA,60+YM*10/MA)
890 NEX=0 :: FOR D=DA TO DE
STEP S*MA/10
900 CALL KEY(0,T,ST):: IF T=
85 THEN 940
910 CALL LINK("SETTO",10*X(D
)/MA,10*Y(D)/MA):: IF LI$<>"
AUS" THEN O6=D :: D=D+S*MA/1
0 :: CALL LINK("MOVETO",10*X
(D)/MA,10*Y(D)/MA):: D=O6
920 NEXT D :: NEX=1
930 GOTO 970
940 CALL SOUND(200,500,5)
950 CALL WAIT(200)
960 GOTO 980
970 CALL SOUND(200,440,3)
980 CALL KEY(0,T,ST):: IF ST
=0 THEN 980 :: IF NEX=0 THEN
 IF T=67 THEN 900
```

```
990 CALL LINK("CLSCRN")
1000 CALL LINK("BYEBYE")
1010 CALL CLEAR :: CALL SCRE
EN(2):: NEX=1
1020 PRINT "FOLGENDES KANN M
AN AENDERN:": :" 1 = INTERVA
LL (";DA;",";DE;")": :" 2 =
MASSTAB (";MA;")": :
1030 PRINT " 3 = MITTL. X-WE
RT (";XM;")": :" 4 = MITTL.
Y-WERT (";YM;")": :" 5 = SCH
RITTLAENGE (";S;")": :
1040 PRINT " 6 = KURVE ZEICH
NEN": :" 7 = ACHSENKREUZ (";
A$;")": :" 8 = LINIENVERB. (
";LI$;")": :" 9 = FUNKTION":
 :
1050 PRINT " N = NULLSTELLEN
": :" E = EXTREMWERTE": :" W
 = WENDEPUNKTE"
1060 CALL KEY(0,T,ST):: IF S
T=0 THEN 1060 ELSE IF T=48 T
HEN CALL CLEAR :: END
1070 IF T<49 OR T>57 AND T<>
78 AND T<>69 AND T<>87 THEN
1060
1080 IF T=78 THEN 1390
1090 IF T=69 THEN 1520 ELSE
IF T=87 THEN 1680
1100 IF T=49 THEN ACCEPT AT(
2,6):DA :: ACCEPT AT(2,12):D
E
1110 IF T=54 OR T=56 OR T=57
 THEN 1160
1120 IF T=55 THEN IF A$="EIN
" THEN A$="AUS" ELSE A$="EIN
" ELSE IF T<>49 THEN 1150 EL
SE 1170
1130 DISPLAY AT(13,19)SIZE(-
4):A$;")"
1140 GOTO 1170
1150 ACCEPT AT((T-48)*2,6):W
1160 ON T-48 GOTO 1170,1180,
1190,1200,1210,830,1170,1360
,1220
1170 GOTO 1060
1180 MA=W :: GOTO 1170
1190 XM=W :: GOTO 1170
1200 YM=W :: GOTO 1170
1210 S=W :: GOTO 1170
1220 CALL CLEAR
1230 PRINT "GEBEN SIE BEIM E
RSCHEINEN": :"DES CURSORS ZU
ERST DEN WERT": :"FUER 'X' E
IN.DANACH": :"GEBEN SIE DEN
WERT FUER 'Y'": :
1240 PRINT "EIN.BEI EINER FU
NKTION": :"MUSS DER 'X'-WERT
": :"GLEICH'D' GESETZT WERDE
N.": :"'D' IST IMMER PARAMET
ER."
1250 PRINT :"DIE FUNKTIONEN
HEISSEN:": :"X=";X$: :"Y=";Y
$
1260 PRINT :"AENDERN? (J/N)"
1270 CALL KEY(0,T,ST):: IF T
=78 THEN 1010 ELSE IF T<>74
THEN 1270
1280 PRINT :"1 = 'X'-AENDERN
": :"2 = 'Y'-AENDERN": :"3 =
 BEIDES"
1290 CALL KEY(0,T,ST):: IF T
=49 THEN ZZ=300 :: HIL=0 ELS
E IF T=50 THEN ZZ=310 ELSE I
F T=51 THEN ZZ=300 :: HIL=1
ELSE 1290
1300 GOTO 340
1310 ZZ=300 :: GOTO 320
1320 CALL CLEAR
1330 CALL ERR(AAA,BBB,CCC,DD
D)
1340 PRINT "*ERROR #";AAA;"I
N ZEILE";DDD: : : :
1350 END
1360 IF LI$="EIN" THEN LI$="
AUS" ELSE LI$="EIN"
1370 DISPLAY AT(15,19)SIZE(3
):LI$
1380 GOTO 1060
1390 ! NULLSTELLENBERECHNUNG
1400 CALL CLEAR
1410 PRINT "BERECHNUNG DER N
ULLSTELLEN": :"DER Y-FUNKTIO
N DER": :"PARAMETERGLEICHUNG
EN": :
1420 DD=DA
1430 E=(DE-DA)/20 :: H=1E-5
1440 IF SGN(Y(DD))=SGN(Y(DD+
E))THEN DD=DD+E :: IF DD>DE
THEN 1780 ELSE 1440
1450 IF ABS(Y(DD+E))>=H THEN
 1470
1460 DD=DD+E :: GOSUB 1500 :
: GOTO 1430
1470 IF ABS(Y(DD))>=H THEN 1
490
1480 GOSUB 1500 :: GOTO 1430
1490 E=E/2 :: GOTO 1440
1500 PRINT :: PRINT :: DISPL
AY AT(23,1):USING "NULLSTELL
EN BEI X = ###.####":X(DD)
1510 DD=DD+H :: RETURN
1520 REM
1530 CALL CLEAR
1540 PRINT "BERECHNUNG DER E
XTREMWERTE": :"DER Y-FUNKTIO
N DER": :"PARAMETERGLEICHUNG
EN": :
```

```
1550 DEF F(D)=(Y(D+H)-Y(D))/
(X(D+H)-X(D))
1560 DD=DA
1570 E=(DE-DA)/20 :: H=1E-5
1580 IF SGN(F(DD))=SGN(F(DD+
E))THEN DD=DD+E :: IF DD>DE
THEN 1780 ELSE 1580
1590 IF ABS(F(DD+E))>=H THEN
 1620
1600 IF SGN(F(DD))=-1 THEN F
U$="MINIMUM" ELSE FU$="MAXIM
UM"
1610 DD=DD+E :: GOSUB 1660 :
: GOTO 1570
1620 IF ABS(F(DD))>=H THEN 1
650
1630 IF SGN(F(DD+E))=1 THEN
FU$="MINIMUM" ELSE FU$="MAXI
MUM"
1640 GOSUB 1660 :: GOTO 1570
1650 E=E/2 :: GOTO 1580
1660 PRINT :: PRINT :: DISPL
AY AT(23,1):USING FU$&" BEI
X = ###.####":X(DD)
1670 DD=DD+H :: RETURN
1680 REM
1690 CALL CLEAR
1700 PRINT "BERECHNUNG DER W
ENDEPUNKTE": :"DER Y-FUNKTIO
N DER": :"PARAMETERGLEICHUNG
EN": :
1710 DEF FF(D)=(Y(D+2*H)-2*Y
(D+H)+Y(D))/(X(D+H)-X(D))^2
1720 DD=DA
1730 E=(DE-DA)/20 :: H=2E-4
1740 IF SGN(FF(DD))=SGN(FF(D
D+E))THEN DD=DD+E :: IF DD>D
E THEN 1780 ELSE 1740
1750 IF ABS(FF(DD+E))<=H THE
N PRINT :: DISPLAY AT(23,1):
USING "WENDEPUNKT BEI X = ##
#.###":X(DD+E):: DD=DD+E+H :
: GOTO 1730
1760 IF ABS(FF(DD))<=H THEN
PRINT :: DISPLAY AT(23,1):US
ING "WENDEPUNKT BEI X = ###.
###":X(DD):: DD=DD+H :: GOTO
 1730
1770 E=E/2 :: GOTO 1740
1780 PRINT :: PRINT "BERECHN
UNG BEENDET !"
1790 CALL WAIT(4000):: GOTO
1010
```

BÖRSE

HILFE! Suche ganzes TI-99/4A Gehäuse für ca. 15 bis 20 DM. Angebote an Michael v.d. Mühlen, Tel. 02103/44144

```
100 ! **********************
110 ! *                    *
120 ! *     Baustatik      *
130 ! *   Copyright by     *
140 ! *    J. Picksak      *
150 ! *                    *
160 ! *   Bearbeitet vom   *
170 ! *      Team des      *
180 ! *  Aktuell Verlages  *
190 ! *      Muenchen      *
200 ! *                    *
210 ! * Benoetigte Geraete *
220 ! *  TI99/4A  Konsole  *
230 ! *     Ext. Basic     *
240 ! * Drucker (optional) *
250 ! *                    *
270 ! *  Speicherbelegung  *
280 ! *     4326 Bytes     *
290 ! **********************
295 !
300 CALL CLEAR
310 DISPLAY TAB(10);"COPYRIG
HT BY"
320 PRINT
330 DISPLAY TAB(10);"JOH.FR.
PICKSAK"
340 DISPLAY TAB(10);"CIRKSEN
ASTRASSE 3"
350 DISPLAY TAB(10);"TEL.049
24/995"
360 DISPLAY TAB(10);"2956 OL
DERSUM"
370 PRINT
380 DISPLAY TAB(10);"BITTE W
ARTEN!"
390 FOR PAUSE=1 TO 1000 :: N
EXT PAUSE
400 CALL CLEAR
410 DISPLAY AT(10,5)BEEP:"DU
RCHBIEGUNG VON"
420 DISPLAY AT(12,5):"EINFEL
DTRAEGERN"
430 DISPLAY AT(13,5):"======
=========="
440 PRINT
450 DISPLAY AT(15,5):"GELENK
IG GELAGERT";TAB(26);"1"
460 DISPLAY AT(17,5):"BEIDSE
IT. EINGESP.";TAB(26);"2"
470 DISPLAY AT(19,10):"E N D
 E";TAB(26);"3"
480 DISPLAY AT(23,7):"BITTE
WAEHLEN !"
490 ! WAEHLEN
500 CALL KEY(0,R,ST)
510 IF ST=0 THEN 500
520 IF R<48 THEN 500
530 IF R>51 THEN 500
540 ON R-48 GOTO 550,1300,16
10
```

```
550 CALL CLEAR
560 DISPLAY AT(1,4):"TRAEGER
 GEL.GELAGERT"
570 DISPLAY AT(2,4):"=======
=============="
580 DISPLAY AT(4,2):"SPANNWE
ITE =" :: ACCEPT AT(4,20):L
590 DISPLAY AT(6,2):"TRAEGHE
ITSM.(CM^4)" :: ACCEPT AT(6,
20):IX
600 DISPLAY AT(8,2):"GLEICHL
AST (KN/M)" :: ACCEPT AT(8,2
0):Q
610 !
620 DISPLAY AT(15,0):"DURCHB
IGUNG INF.GLEICHLAST"
630 FQM=(L^4*Q*0.62)/IX :: F
QM=INT(FQM*100+0.5)/100
640 DISPLAY AT(16,2):"------
----------------------"
650 DISPLAY AT(18,10):"F(Q)=
";FQM;"CM"
660 DISPLAY AT(20,2):"======
======================"
670 DISPLAY AT(22,2):"Y INFO
LGE EINZELLASTEN?"
680 DISPLAY AT(24,12):"-J/N-
" :: ACCEPT AT(24,22):O$
690 IF (O$="J")+(O$="j")THEN
 740
700 IF (O$="N")+(O$="n")THEN
 710
710 FKF=0
720 CALL CLEAR
730 GOTO 990
740 CALL CLEAR
750 DISPLAY AT(2,2):"ANZ. DE
R EINZELLASTEN" :: ACCEPT AT
(2,25):ANZ
760 FKF=0
770 FOR I=1 TO ANZ
780 DISPLAY AT(6,8):"F=" ::
ACCEPT AT(6,12):F
790 DISPLAY AT(8,8):"A=" ::
ACCEPT AT(8,12):A
800 LET B=L-A
810 IF A<L/2 THEN 840
820 IF A>L/2 THEN 830
830 A=L-A :: B=L-A
840 MO=L/4 :: MFM=(F*B/L)*A
850 DELTA=0.5*MO*MFM*L/B
860 EIY=(L/2)-(A^2)*2/(3*L)
870 FK(I)=100*(DELTA*EIY)/(2
.1*IX)
880 FKF=FKF+FK(I)
890 NEXT I
900 FKF=INT(FKF*100+0.5)/100
910 LET FQMFKF=FQM+FKF
920 CALL CLEAR
930 DISPLAY AT(6,2):"DURCHBI
EGUNG IN FELDMITTE"
940 DISPLAY AT(8,10):"Y(Q)="
;FQM;"CM"
950 DISPLAY AT(9,10):"Y(F)="
;FKF;"CM"
960 DISPLAY AT(10,10):"-----
----------------------"
970 DISPLAY AT(12,7):"SUMME
F=";FQMFKF
980 DISPLAY AT(13,2):"======
========================"
990 DISPLAY AT(15,1):"SOLL A
USGEDRUCKT WERDEN?J/N"
1000 CALL KEY(0,T,ST):: IF S
T=0 THEN 1000
1010 IF (T<>74 OR T<>78 OR T
<>106 OR T<>110)=-4 THEN 100
0
1020 IF T=78 OR T=110 THEN 1
290
1030 DISPLAY AT(17,2):"DRUCK
EROPTIONEN"
1040 POPT$=""
1050 INPUT POPT$
1060 OPEN #1:POPT$,OUTPUT
1070 PRINT #1:TAB(14);CHR$(1
4);"DURCHBIEGUNG IN FELDMITT
E"
1080 PRINT #1:TAB(14);"=====
============================
=================="
1090 PRINT #1 ::
1100 PRINT #1:TAB(14);"EINFE
LDTRAEGER LAGERUNG:GELENKIG
ODER EINGESPANNT."
1110 PRINT #1:
1120 PRINT #1:TAB(14);"TRAEG
HEITSMOMENT IN CM^4=";IX
1130 PRINT #1:TAB(14);"SPANN
WEITE IN METER=";L
1140 PRINT #1:
1150 PRINT #1:TAB(14);"LASTE
N SIEHE STATIKSKIZZE"
1160 PRINT #1:TAB(14);"-----
----------------------------
---------"
1170 PRINT #1:TAB(14);"Y (Q)
 IN FELDMITTE =";TAB(40);FQM
;"CM"
1180 PRINT #1:TAB(14);"Y (F)
      -//-     =";TAB(40);FKF
;"CM"
1190 PRINT #1:TAB(14);"-----
----------------------------
---------"
1200 LET FQMFKF=FQM+FKF
1210 PRINT #1:TAB(14);"SUMME
 Y             =";TAB(40);FQM
FKF;"CM"
1220 PRINT #1:TAB(14);"=====
============================
```

```
=========" 
1230 PRINT #1:
1240 LET VZL=L*100/FQMFKF
1250 VZL=INT(VZL*100+0.5)/10
0
1260 PRINT #1:TAB(14);"DURCH
BIEGUNG ENTSPRICHT  1:";VZL;
"DER SPANNWEITE."
1270 PRINT #1:TAB(14);"-----
----------------------------
-----------------"
1280 CLOSE #1
1290 GOTO 400
1300 ! EINGESP. TRAEGER
1310 CALL CLEAR
1320 DISPLAY AT(1,4):"EINGES
P. TRAEGER"
1330 DISPLAY AT(2,4):"======
=========="
1340 DISPLAY AT(4,2):"SPANNW
EITE=" :: ACCEPT AT(4,20):L
1350 DISPLAY AT(6,2):"TRAEGH
EITSM.(CM^4)=" :: ACCEPT AT(
6,20):IX
1360 DISPLAY AT(8,2):"GLEICH
LAST (KN/M)=" :: ACCEPT AT(8
,20):Q
1370 DISPLAY AT(15,0):"DURCH
BIEGUNG INF.GLEICHLAST"
1380 FQM=(L^4*Q*0.12)/IX ::
FQM=INT(FQM*100+0.5)/100
1390 DISPLAY AT(16,2):"-----
-----------------------"
1400 DISPLAY AT(18,10):"Y(Q)
=";FQM;"CM"
1410 DISPLAY AT(20,2):"=====
======================"
1420 DISPLAY AT(22,2):"Y INF
OLGE EINZELLASTEN?"
1430 DISPLAY AT(24,12):"-J/N
-" :: ACCEPT AT(24,22):OF$
1440 IF OF$="J" OR OF$="j" T
HEN 1470
1450 IF OF$="N" OR OF$="n" T
HEN 1460
1460 CALL CLEAR :: GOTO 990
1470 ! EINZELLASTEN
1480 CALL CLEAR
1490 DISPLAY AT(2,2):"ANZ.DE
R EINZELLASTEN?" :: ACCEPT A
T(2,24):ANZ
1500 FKF=0 :: FQMFKF=0
1510 FOR I=1 TO ANZ
1520 DISPLAY AT(6,8):"F=" ::
 ACCEPT AT(6,12):F
1530 DISPLAY AT(8,8):"A=" ::
 ACCEPT AT(8,12):A
1540 LET B=L-A
1550 LFL=(A^2*B^3)/(3*L-2*A)
^2
1560 LFR=(2*F*100)/(3*2.1*IX
)
1570 FK(I)=LFL*LFR
1580 FKF=FKF+FK(I)
1590 NEXT I
1600 GOTO 900
1610 STOP
```

BUDDY BALLOON

Ist ein Geschicklichkeitsspiel, das die ganze Konzentration in Anspruch nimmt. Es besteht aus 4 verschiedenen Bildern, die jeweils 3mal durchlaufen werden müssen. Am Anfang stehen 4 Spielfiguren zur Verfügung. Sollten Sie eine Runde nicht schaffen, ist eine Figur verloren. Die Spielgeschwindigkeit nimmt von Runde zu Runde zu, die zur Verfügung stehende Zeit ab.

Die Erklärungen zu den einzelnen Runden stehen in der Spielanleitung am Anfang des Spiels.
Noch ein Tip zu Bild 1: Konzentrieren Sie sich nicht so sehr auf die Punkte, sondern auf den genauen Absprungzeitpunkt, um den nächsten Ballon zu erreichen.
Und nun viel Vergnügen und ein absturzfreies Spiel.

Frajo Fry

BILDMUSTER-GENERATOR

Das Programm „Bildmustergenerator" ist gedacht für Fernseh-Bastler, die sich die Anschaffung eines teuren „echten" Bildmustergenerators sparen möchten. Es ermöglicht die Einstellung von Konvergenz, Bildlinearität, Zeilenlinearität, Kissenentzerrung, Grauabgleich, Farbwiedergabe etc. Welche Arten von Testbildern erzeugt werden können, erklärt das Programm im Menü.

Natürlich bietet das Programm keinen vollwertigen Ersatz für einen richtigen Bildmustergenerator, vor allem durch die Einschränkung des Formats am oberen und unteren Bildrand.
Trotzdem stellt es eine preiswerte Alternative für Besitzer eines Home-Computers dar.

Werner Liebender

WESTERN

Dieses Spiel wird bei einigen, die früher eine Atari-Spielkonsole besessen hatten, sicherlich Erinnerungen wachrufen.
Es ist eine für den TI 99 umgeschriebene Version eines der Atari-Spielmodule. Zwei Revolverhelden treffen sich in der Wüste, um ihr tödliches Duell auszutragen. Als einzige Dekkung steht ein Kaktus zwischen ihnen. Zwei Spieler steuern über Joystick die Revolverhelden in jede beliebige Richtung des Bildschirms. Durch den Feuerknopf und Drücken des

Bitte lesen Sie weiter auf Seite 35

```
100 ! *********************
105 ! *                   *
110 ! *   BUDDY BALLOON   *
120 ! *   Copyright by    *
130 ! *    Frajo  Fry     *
140 ! *                   *
150 ! *   FUER TI/99/4A   *
160 ! *   & Ext. Basic    *
162 ! *                   *
164 ! * Speicherbelegung: *
166 ! *    11089 Bytes    *
168 ! *                   *
170 ! *********************
175 !
180 !   *** Einleitung ***
182 !
190 CALL CHAR(64,"3C4299A1A1
99423C"):: CALL CLEAR :: DIS
PLAY AT(11,9):"BUDDY BALLOON
" :: DISPLAY AT(13,8):"@ BY
 FRAJO FRY"
200 FOR I=1 TO 3 :: FOR J=11
0 TO 1100 STEP 110 :: CALL S
OUND(100,J,0):: NEXT J :: FO
R J=1100 TO 110 STEP -110 ::
 CALL SOUND(100,J,0):: NEXT
J
210 FOR J=110 TO 1100 STEP 1
10 :: CALL SOUND(100,J,0,121
0-J,0):: NEXT J :: NEXT I ::
 DISPLAY AT(24,4)BEEP:"SPIEL
ERKLAERUNG  (J/N)?"
220 CALL KEY(0,K,S):: IF S=0
 THEN 220 :: IF K=78 THEN 36
0 ELSE IF K<>74 THEN 220
230 CALL CLEAR :: DISPLAY AT
(1,1):"1. BILD: DU MUSST VON
 BALLON": :"ZU BALLON SPRING
EN UND MOEG-": :"LICHST VIEL
E PUNKTE TREFFEN."
240 DISPLAY AT(8,1):"SPRUNGT
ASTE: >SPACE<" :: GOSUB 1500
250 DISPLAY AT(1,1):"2. BILD
: DU MUSST IN JEDER": :"ETAG
E DIE 3 BALLONS BERUEH-": :"
REN UND DANN ZUR TREPPE ZU-"
: :"RUECKGEHEN. WENN DU DEN
BAL-": :
260 DISPLAY AT(9,1):"LON GAN
Z OBEN BERUEHRST,": :"SCHWEB
ST DU MIT IHM ZUR ERDE": :"Z
URUECK. EIN KONTAKT MIT DEN"
: :"WEISSEN KUGELN IST ZU VE
R-": :
270 DISPLAY AT(17,1):"MEIDEN
.": : :"SPRUNGTASTE: >SPACE<
": :"BEWGUNGSTASTEN: >S< & >
D<" :: GOSUB 1500
280 DISPLAY AT(1,1):"3. BILD
: DU MUSST IN DER 1.": :"UND
 3. ETAGE IM SPRUNG DEN": :"
BLAUEN, IN DER 2. ETAGE DEN"
: :"ROTEN BALLON BERUEHEN, U
M": :
290 DISPLAY AT(9,1):"EINE ET
AGE HOEHER ZU GELAN-": :"GEN
. WICHTIG IST EIN GENAUES":
:"AN- UND ABSPRINGEN. ERST":
 :"GANZ OBEN GEHST DU ZUR TR
EP-": :"PE ZURUECK."
300 DISPLAY AT(20,1):"AN- UN
D ABSPRUNG: >SPACE<": :"BEWE
GUNG: >S< & >D<" :: GOSUB 15
00
310 DISPLAY AT(1,1):"4. BILD
: DU MUSST MIT PFEI-": :"LEN
 AUF DEN BALLON SCHIESSEN":
:"UND MOEGLICHST DEN HELLEN"
: :"PUNKT TREFFEN. DADURCH":
 :
320 DISPLAY AT(9,1):"STEIGST
 DU DIE LEITER HINAB.": :"DA
NACH GEHT ES WIEDER MIT": :"
DEM 1. BILD WEITER.": : :"SC
HUSSTASTE: >SPACE<" :: GOSUB
 1500
330 DISPLAY AT(1,1):"DU HAST
 AM ANFANG 4 BUDDYS.": :"DAS
 SPIEL BESTEHT AUS 3 RUN-":
:"DEN. DIE SCHNELLIGKEIT": :
"NIMMT JEWEILS ZU, DIE ZUR":
 :
340 DISPLAY AT(9,1):"VERFUEG
UNG STEHENDE ZEIT AB.": :"NA
CH JEDEM BILD GIBT ES DIE":
:"RESTZEIT (*RUNDE) ALS BONU
S.": :"SCHAFFST DU ALLE 12 B
ILDER,": :
350 DISPLAY AT(17,1):"GIBT E
S FUER JEDEN UEBRIG-": :"GEB
LIEBENEN BUDDY EINEN": :"SUP
ERBONUS VON 1000 PUNKTEN.":
:"VIEL SPASS BEI BUDDY BALLO
ON" :: GOSUB 1500
360 CALL CLEAR :: CALL SCREE
N(11):: CALL MAGNIFY(3):: RA
NDOMIZE :: M=4 :: P=0 :: R=1
 :: T=401 :: Z=11
370 CALL CHAR(96,"070F0F1E1A
1A0A04070F17270707080 6C0E01C
420E10204080F8C0E0E0C0502")
380 CALL CHAR(100,"030738427
00804020 11F030707030A04E0F0F
0785858502 0E0F0E8E4E0E0106")
390 CALL CHAR(104,"03070F0D0
F0F0F0F0703000000000007C0E0F
0F0F0F0F0F0E0C080402020C")
400 CALL CHAR(108,"00000000
B1A2A4A0A0A0A0B0000000000000
000DE525252525252DE")
```

```
410 CALL CHAR(112,"00000007
B0A0A7A4242427B0000000000000
000DE52525252525252DE")
420 CALL CHAR(116,"00000007
B0A0A7A0A0A0A7B0000000000000
000DE52525252525252DE")
430 CALL CHAR(120,"000000000
00000003070F0F0F0F070300000
00000000000C0E0F0F0F0F0E0C")
440 CALL CHAR(124,RPT$("0",4
6)&"0402FF0204",128,"3844483
07CB0306C")
450 CALL CHAR(136,"E0E0E0FFF
FE0E0E",137,"070707FFFF07070
7",138,"000000FF55FFAAFF")::
 CALL COLOR(14,13,1)
460 CALL HCHAR(24,4,138,29):
: CALL VCHAR(4,2,136,21):: C
ALL VCHAR(4,3,137,21)
470 CALL HCHAR(1,2,90):: CAL
L HCHAR(1,10,80):: CALL HCHA
R(1,20,72):: GOSUB 1450
480 ! 1. BILD
490 Z=Z-5 :: FOR I=1 TO 19 S
TEP 6 :: FOR J=1 TO 3 :: CAL
L SPRITE(#I+J,104,(I+2)/3+2,
I*8+1,J*85-20):: NEXT J :: N
EXT I
500 FOR I=1 TO 13 STEP 6 ::
FOR J=1 TO 3 :: CALL SPRITE(
#3+I+J,108,16,I*8+25,J*85-55
):: NEXT J :: NEXT I
510 A=1 :: CALL MOTION(#2,0,
Z,#3,0,Z,#4,0,Z,#5,0,Z+3,#6,
0,Z+3,#7,0,Z+3)
520 CALL MOTION(#8,0,-Z,#9,0
,-Z,#10,0,-Z,#11,0,-Z-4,#12,
0,-Z-4,#13,0,-Z-4)
530 CALL MOTION(#14,0,Z,#15,
0,Z,#16,0,Z,#17,0,Z+5,#18,0,
Z+5,#19,0,Z+5)
540 CALL MOTION(#20,0,-Z,#21
,0,-Z,#22,0,-Z)
550 CALL START(Y):: CALL MOT
ION(#1,-6,0)
560 CALL POSITION(#1,X,Y)::
IF X>21 THEN 560 :: CALL MOT
ION(#1,0,0)
570 FOR I=2 TO 4 :: CALL COI
NC(#1,#I,12,C):: IF C THEN C
ALL MOTION(#1,0,Z):: GOTO 59
0
580 NEXT I :: GOTO 570
590 IF A/2=INT(A/2)THEN CALL
 PATTERN(#1,100)ELSE CALL PA
TTERN(#1,96)
600 CALL KEY(0,K,S):: CALL T
IME(T):: IF T=0 THEN T=801+R
*150-ABS(Z)*100 :: GOTO 690
610 IF K<>32 THEN 600 :: CAL
L SOUND(-50,500,0):: CALL MO
TION(#1,6,Z/2)
620 CALL POSITION(#1,X,Y)::
IF X>20+A*48 THEN 690
630 IF X<166 THEN 650 :: Z=A
BS(Z)+1 :: CALL MOTION(#1,0,
0):: CALL LOCATE(#1,172,Y)::
 IF Z=R*2+7 THEN 710
640 FOR I=1 TO 13 STEP 6 ::
FOR J=1 TO 3 :: CALL PATTERN
(#3+I+J,92+(Z-2*R)*4):: NEXT
 J :: NEXT I :: GOTO 510
650 FOR I=A*6-1 TO A*6+4 ::
CALL COINC(#1,#I,10,C):: IF
C THEN 670
660 NEXT I :: GOTO 620
670 IF I>A*6+1 THEN 680 :: C
ALL SOUND(100,1000,0):: CALL
 DELSPRITE(#I):: P=P+ABS(Z*1
00)-300-R*200 :: GOTO 620
680 CALL POSITION(#I,X1,Y1):
: CALL LOCATE(#1,21+A*48,Y1)
:: Z=-Z :: CALL MOTION(#1,0,
Z):: CALL POINTS(P):: A=A+1
:: GOTO 590
690 CALL FALL :: GOSUB 1440
:: GOTO 510
700 ! 2. BILD
710 GOSUB 1460 :: GOSUB 1490
 :: FOR I=6 TO 18 STEP 6 ::
CALL HCHAR(I,4,138,29):: NEX
T I
720 FOR I=1 TO 3 :: FOR J=2
TO 8 STEP 3 :: CALL SPRITE(#
I+J,104,I*2+3,J*16+17,I*80):
: NEXT J :: NEXT I
730 FOR I=1 TO 3 :: FOR J=1
TO I :: CALL SPRITE(#11+I*J,
120,16,220-I*48,J*(200-I*40)
):: NEXT J :: NEXT I
740 CALL SPRITE(#2,104,3,15,
120,0,Z)
750 CALL START(Y):: IF A=4 T
HEN 780
760 CALL MOTION(#1,-6,0)
770 CALL POSITION(#1,X1,Y1):
: IF X1>A*48-17 THEN 770
780 CALL MOTION(#1,0,0):: CA
LL LOCATE(#1,A*48-20,14)
790 CALL MOTION(#12,0,Z+2,#1
3,0,Z,#15,0,Z,#14,0,Z-2,#17,
0,Z-2,#20,0,Z-2)
800 XX=X :: CALL KEY(0,K,S):
: CALL TIME(T):: IF T=0 THEN
 T=151+(A-R)*50 :: GOTO 910
810 IF K<>32 OR A=1 THEN 870
820 FOR I=-24 TO 0 STEP 3 ::
 CALL MOTION(#1,I,XX):: NEXT
 I
830 FOR I=A*3-3 TO A*3-1 ::
CALL COINC(#1,#I,13,C):: IF
C=0 THEN 850 :: CALL PATTERN
```

```
(#I,124-A*4):: W=W+1
84Ø P=P+5ØØ-A*1ØØ :: CALL PO
INTS(P):: CALL SOUND(1ØØ,44Ø
,Ø,66Ø,2,88Ø,4):: CALL DELSP
RITE(#I)
85Ø NEXT I :: FOR I=Ø TO 24
STEP 3 :: CALL MOTION(#1,I,X
X):: NEXT I :: CALL MOTION(#
1,Ø,Ø)
86Ø CALL POSITION(#1,X1,Y1):
: CALL LOCATE(#1,A*48-2Ø,Y1)
:: GOTO 9ØØ
87Ø X=Ø :: IF K=83 THEN X=-4
 ELSE IF K=68 THEN X=4
88Ø IF X=Ø THEN 89Ø :: IF X=
4 THEN CALL PATTERN(#1,96)EL
SE CALL PATTERN(#1,1ØØ)
89Ø CALL MOTION(#1,Ø,X*2)::
CALL POSITION(#1,X1,Y1):: IF
 Y1>242 THEN 91Ø :: IF Y1>13
 THEN 9ØØ :: IF W<3 THEN 91Ø
 :: W=Ø :: A=A-1 :: GOTO 76Ø
9ØØ CALL COINC(ALL,C):: IF C
=Ø THEN 8ØØ :: IF A=1 THEN 9
2Ø
91Ø CALL DELSPRITE(#2):: FOR
 I=12 TO 2Ø :: CALL MOTION(#
I,Ø,Ø):: NEXT I :: CALL FALL
 :: GOSUB 144Ø :: GOTO 74Ø
92Ø CALL MOTION(#1,Ø,Ø,#2,Ø,
Ø):: CALL POSITION(#2,X1,Y1)
:: CALL LOCATE(#1,X1+16,Y1)
93Ø CALL KEY(Ø,K,S):: CALL T
IME(T):: IF T=Ø THEN T=1Ø1-R
*2Ø :: GOTO 91Ø
94Ø IF K=83 THEN X=-4 ELSE I
F K=68 THEN X=4
95Ø CALL MOTION(#1,Z-2,X,#2,
Z-2,X):: CALL COINC(ALL,C)::
 IF C THEN 91Ø
96Ø CALL POSITION(#1,X1,Y1):
: IF X1<167 THEN 93Ø
97Ø CALL MOTION(#2,Ø,Ø,#1,Ø,
Ø):: CALL POSITION(#1,X,Y)::
 CALL LOCATE(#1,172,Y)
98Ø CALL PATTERN(#2,116):: P
=P+3ØØ :: CALL SOUND(25Ø,8ØØ
,Ø):: CALL POINTS(P):: CALL
WAIT(1ØØ)
99Ø GOSUB 146Ø :: GOSUB 149Ø
1ØØØ ! 3. BILD
1Ø1Ø FOR I=4 TO 14 :: CALL S
PRITE(#I,12Ø,16,I*8-15,I*16-
1Ø):: NEXT I :: CALL SPRITE(
#2,1Ø4,7,76,31,#3,1Ø4,5,124,
235)
1Ø2Ø CALL START(Y):: IF A=1
THEN 1Ø4Ø :: CALL MOTION(#1,
-6,Ø)
1Ø3Ø CALL POSITION(#1,X,Y)::
 IF X>223-A*48 THEN 1Ø3Ø
1Ø4Ø CALL MOTION(#1,Ø,Ø):: C
ALL LOCATE(#1,22Ø-A*48,24)
1Ø5Ø FOR I=4 TO 14 :: CALL M
OTION(#I,RND*Z+I,Ø):: NEXT I
 :: CALL MOTION(#2,-Z-2,Ø,#3
,-Z-2,Ø)
1Ø6Ø CALL KEY(Ø,K,S):: CALL
TIME(T):: IF T=Ø THEN T=4Ø1-
(A+R)*5Ø :: GOTO 12ØØ
1Ø7Ø IF K<>32 THEN 116Ø :: I
F A=2 THEN V=2 ELSE V=3
1Ø8Ø FOR I=-24 TO Ø STEP 3 :
: CALL MOTION(#1,I,Ø):: NEXT
 I :: IF A=4 THEN 1Ø9Ø :: CA
LL COINC(#1,#V,15-R,C):: IF
C THEN 111Ø
1Ø9Ø FOR I=Ø TO 24 STEP 3 ::
 CALL MOTION(#1,I,Ø):: NEXT
I :: CALL MOTION(#1,Ø,Ø)
11ØØ CALL POSITION(#1,X1,Y1)
:: CALL LOCATE(#1,22Ø-A*48,Y
1):: GOTO 119Ø
111Ø CALL MOTION(#V,Ø,Ø):: C
ALL POSITION(#V,X2,Y2):: CAL
L LOCATE(#1,X2+12,Y2):: CALL
 MOTION(#V,-Z+2,Ø,#1,-Z+2,Ø)
112Ø CALL KEY(Ø,K,S):: CALL
TIME(T):: IF T=Ø THEN T=4Ø1-
(A+R)*5Ø :: GOTO 12ØØ
113Ø IF K<>32 THEN 112Ø :: C
ALL MOTION(#1,Ø,Ø,#V,-Z-2,Ø)
:: CALL POSITION(#1,X1,Y1)::
 W=ABS(X1-172+A*48):: IF W>2
 THEN 12ØØ
114Ø CALL PATTERN(#V,116-W*4
):: P=P+3ØØ-W*1ØØ :: CALL SO
UND(5ØØ,88Ø-W*22Ø,Ø):: CALL
POINTS(P)
115Ø CALL LOCATE(#1,172-A*48
,Y1):: CALL PATTERN(#V,1Ø4):
: A=A+1 :: GOTO 1Ø6Ø
116Ø X=Ø :: IF K=83 THEN X=-
4 ELSE IF K=68 THEN X=4
117Ø IF X=Ø THEN 118Ø :: IF
X=4 THEN CALL PATTERN(#1,96)
ELSE CALL PATTERN(#1,1ØØ)
118Ø CALL MOTION(#1,Ø,X*2)::
 CALL POSITION(#1,X1,Y1):: I
F Y1>242 THEN 12ØØ :: IF Y1<
24 THEN IF A=4 THEN 121Ø ELS
E 12ØØ
119Ø CALL COINC(ALL,C):: IF
C=Ø THEN 1Ø6Ø
12ØØ FOR I=2 TO 14 :: CALL M
OTION(#I,Ø,Ø):: NEXT I :: CA
LL FALL :: GOSUB 144Ø :: GOT
O 1Ø2Ø
121Ø CALL MOTION(#1,Ø,Ø):: C
ALL PATTERN(#1,96):: X=28 ::
```

```
 CALL LOCATE(#1,X,13):: CALL
 PATTERN(#2,116,#3,116)
122Ø P=P+6ØØ :: CALL SOUND(2
5Ø,8ØØ,Ø):: CALL POINTS(P)::
 CALL WAIT(5ØØ)
123Ø FOR I=6 TO 18 STEP 6 ::
 CALL HCHAR(I,4,32,29):: NEX
T I :: GOSUB 146Ø :: GOSUB 1
49Ø
124Ø ! 4. BILD
125Ø CALL SPRITE(#2,1Ø4,INT(
4*RND)*2+3,24Ø,INT(RND*16Ø)+
4Ø)
126Ø CALL POSITION(#2,X1,Y1)
:: IF X1+15<X THEN CALL MOTI
ON(#2,-Z*1.5,Ø)ELSE CALL MOT
ION(#2,-Z,Ø)
127Ø CALL KEY(Ø,K,S):: CALL
TIME(T):: IF T>Ø THEN 13ØØ :
: T=451-X*2.5-R*1Ø :: CALL F
ALL :: CALL DELSPRITE(#2)::
GOSUB 144Ø
128Ø CALL START(Y):: CALL MO
TION(#1,-6,Ø)
129Ø CALL POSITION(#1,X1,Y1)
:: IF X1>X THEN 129Ø :: CALL
 MOTION(#1,Ø,Ø):: CALL LOCAT
E(#1,X,13):: GOTO 125Ø
13ØØ IF K<>32 THEN 126Ø :: C
ALL MOTION(#2,Ø,Ø):: CALL PO
SITION(#2,X1,Y1)
131Ø FOR I=Ø TO 15 :: CALL S
OUND(-2Ø,1ØØØ+I*1ØØ,I):: NEX
T I
132Ø CALL SPRITE(#3,124,2,X,
Y1-16):: W=ABS(X-X1+6):: IF
W>2 THEN CALL DELSPRITE(#3):
: CALL MOTION(#2,-Z,Ø):: CAL
L WAIT(4Ø):: GOTO 127Ø
133Ø CALL SOUND(5Ø,-5,Ø):: C
ALL PATTERN(#2,116-W*4):: P=
P+3ØØ-W*1ØØ :: CALL POINTS(P
):: X=X+24-W*8 :: IF X>172 T
HEN X=172
134Ø CALL LOCATE(#1,X,13)::
CALL DELSPRITE(#2,#3):: IF X
<172 THEN 125Ø
135Ø CALL MOTION(#1,Ø,8)
136Ø CALL POSITION(#1,X,Y)::
 IF Y<25Ø THEN 136Ø :: CALL
DELSPRITE(#1):: GOSUB 146Ø :
: R=R+1 :: IF R<4 THEN GOSUB
 149Ø :: GOTO 49Ø
137Ø ! SPIELENDE
138Ø DISPLAY AT(9,7):"SUPER-
BONUS:";M*1ØØØ :: FOR I=12Ø
TO 9ØØ STEP 1Ø :: CALL SOUND
(-1Ø,I,I/6Ø):: NEXT I :: DIS
PLAY AT(9,7)
139Ø IF P<=H THEN 141Ø :: H=
P :: CALL HIGH(H)
14ØØ FOR I=22Ø TO 88Ø STEP 1
1Ø :: CALL SOUND(2ØØ,I,Ø,I+2
Ø,2,I+4Ø,4):: NEXT I
141Ø CALL DELSPRITE(ALL):: R
ESTORE 152Ø :: FOR I=1 TO 1Ø
 :: READ V :: CALL HCHAR(9,I
+11,V):: CALL SOUND(25Ø,-5,Ø
,11Ø,2):: NEXT I
142Ø DISPLAY AT(15,4)BEEP:"E
IN NEUES SPIEL (J/N)?"
143Ø CALL KEY(Ø,K,S):: IF S=
Ø THEN 143Ø :: IF K=74 THEN
36Ø :: IF K<>78 THEN 143Ø ::
 CALL CLEAR :: END

144Ø Z=ABS(Z):: M=M-1 :: IF
M=Ø THEN 139Ø
145Ø Y=256 :: CALL HCHAR(1,3
Ø,32,3):: CALL HCHAR(1,3Ø,12
8,M-1):: CALL TIME(T):: CALL
 POINTS(P):: CALL HIGH(H)::
RETURN
146Ø FOR I=2 TO 22 :: CALL D
ELSPRITE(#I):: NEXT I :: Z=Z
+1 :: B=T*R :: DISPLAY AT(9,
1Ø):"BONUS:";B
147Ø RESTORE 151Ø :: FOR I=1
 TO 11 :: READ V :: CALL SOU
ND(-1ØØ,V,Ø,V+2,Ø,V+4,Ø):: N
EXT I
148Ø P=P+B :: CALL POINTS(P)
:: CALL WAIT(5Ø):: CALL SOUN
D(-1ØØ,131,Ø,133,Ø,135,Ø)::
DISPLAY AT(9,1Ø):"" :: RETUR
N

149Ø T=451-R*5Ø :: CALL TIME
(T):: RETURN
15ØØ DISPLAY AT(24,8):"TASTE
 DRUECKEN" :: CALL KEY(Ø,K,S
):: IF S=Ø THEN DISPLAY AT(2
4,8):: GOTO 15ØØ ELSE CALL C
LEAR :: RETURN
151Ø DATA 262,22Ø,247,196,22
Ø,175,196,165,175,147,165
152Ø DATA 71,65,77,69,32,32,
79,86,69,82
153Ø SUB START(Y):: CALL SPR
ITE(#1,1ØØ,2,172,Y,Ø,-8)
154Ø CALL POSITION(#1,X,Y)::
 IF Y<7 OR Y>1Ø THEN 154Ø ::
 CALL PATTERN(#1,96)
155Ø SUBEND
156Ø SUB FALL :: CALL MOTION
(#1,1Ø,Ø)
157Ø CALL POSITION(#1,C,D)::
 CALL SOUND(5Ø,C*1Ø,Ø):: IF
C<196 THEN 157Ø
158Ø CALL MOTION(#1,Ø,Ø):: S
UBEND
159Ø SUB POINTS(P):: DISPLAY
 AT(1,9)SIZE(7):P :: SUBEND
```

NUTZEN SIE UNSEREN BEQUEMEN POSTSERVICE



KOMMT REGELMÄSSIG ZU IHNEN INS HAUS

Finden Sie Ihre TI REVUE nicht am Kiosk? Weil sie schon ausverkauft ist? Oder „Euer" Kiosk nicht beliefert wurde? Kein Problem! Für ganze 30 DM liefern wir Euch per Post sechs Hefte ins Haus (Ausland 40 DM). Einfach den Bestellschein auf der nächsten Seite ausschneiden – fotokopieren oder abschreiben, in einen Briefumschlag und ab per Post (Achtung: Porto nicht vergessen). TI REVUE kommt dann pünktlich ins Haus.

WICHTIGE RECHTLICHE GARANTIE!

Sie können diesen Abo-Auftrag binnen einer Woche nach Eingang der Abo-Bestätigung durch den Verlag widerrufen— Postkarte genügt. Ansonsten läuft dieser Auftrag jeweils für sechs Ausgaben, wenn ihm nicht vier Wochen vor Ablauf widersprochen wird, weiter.

DAS SUPER-SONDER-ANGEBOT: PRIVATE KLEINANZEIGEN KOSTENLOS!

Das bietet Ihnen ab sofort die TIREVUE KLEINANZEIGEN SIND KOSTENLOS FÜR PRIVATANBIETER! Suchen Sie etwas, haben Sie etwas zu verkaufen, zu tauschen, wollen Sie einen Club gründen? Coupon ausfüllen, auf Postkarte kleben oder in Briefumschlag stecken und abschicken. So einfach geht das. Wollen Sie das Heft nicht zerschneiden, können Sie den Coupon auch fotokopieren. Oder einfach den Anzeigentext uns so schicken, auf Postkarte oder im Brief. Aber bitte mit Druckbuchstaben oder in Schreibmaschinenschrift!

Und: Einschließlich Ihrer Adresse und/oder Telefonnummer sollten acht Zeilen à 28 Anschläge nicht überschritten werden.

ACHTUNG: WICHTIGER HINWEIS!

Wir veröffentlichen nur Kleinanzeigen privater Inserenten, keine gewerblichen Anzeigen. Die kosten pro Millimeter DM 2,50 plus Mehrwertsteuer!

Wir versenden für Privat-Inserenten keine Beleg-Exemplare!

Chiffre-Anzeigen sind nicht gestattet! Wir behalten uns vor, Anzeigen, die gegen rechtliche, sittliche oder sonstige Gebote verstoßen, abzulehnen!

Anzeigenabdruck in der Reihenfolge ihres Eingangs, kein Rechtsanspruch auf den Abdruck in der nächsten Ausgabe!

Die Insertion ist nicht vom Kauf des Heftes abhängig!

Wir behalten uns vor, Anzeigen, die nicht zum Themenkreis des Heftes – Computer – gehören, nicht abzudrucken oder sie nur insoweit zu berücksichtigen, wie es der Umfang des kostenlosen Anzeitenteils zuläßt.

KASSETTEN-SERVICE

Wollt Ihr nicht mehr seitenlange Listings eintippen? Dann benutzt unseren Kassetten-Service: Für ganze zehn DM — nur Vorkasse, keine Rechnung — keine Nachnahme — erhalten Sie alle angebotenen Listings des jeweiligen Heftes auf Kassette per Post ins Haus. Achtung: Bestellungen ohne Geld werden nicht bearbeitet! Lieferzeit: In der Regel zwei bis drei Wochen! Wichtig: Unsere Kassetten werden digital in einem Profistudio von einem Masterband gezogen, nicht irgendwie per Recorder runtergenudelt. Selbstverständlich volles Umtauschrecht bei Nichtfunktionieren.



ANZEIGENSERVICE

Die große Börse für jeden Zweck in der TI REVUE. Kostenlos für Privat-Inserenten. Spottbillig für gewerbliche Anbieter. Einfach Coupon ausschneiden, fotokopieren o.ä., ausfüllen und ab die Post — Freimachen nicht vergessen! — Unsere Adresse steht auf dem Coupon, ebenso die Preise für gewerbliche Anbieter! Achtung! Wir weisen ausdrücklich darauf hin, daß wir offensichtlich gewerbliche Anzeigen nicht kostenlos veröffentlichen und uns jedweden Abdruck kostenloser Anzeigen vorbehalten müssen, insbesondere, wenn deren Inhalt nicht TI-typisch ist oder gegen geltendes Recht verstößt. Private Chiffreanzeigen werden nicht aufgenommen. Für Privatanbieter: maximal acht Zeilen à 28 Anschläge.

TI REVUE
Anzeigenabteilg.
Postfach 1107
8044 Lohhof

Name ______
Vorname ______
Straße/Hausnr. ______
PLZ/Ort ______

ABO SERVICE-KARTE

9

TI

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

Coupon

Ja, ich möchte von Ihrem Angebot Gebrauch machen.

Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten sechs Ausgaben an untenstehende Anschrift.

Name ______
Vorname ______
Straße/Hausnr. ______
Plz/Ort ______
Ich bezahle:
☐ per beiliegendem Verrechnungsscheck
☐ gegen Rechnung
☐ bargeldlos per Bankeinzug von meinem Konto
bei (Bank) und Ort ______
Kontonummer ______
Bankleitzahl ______
(steht auf jedem Kontoauszug)
Unterschrift ______

Von meinem Widerspruchsrecht habe ich Kenntnis genommen.

Unterschrift ______

TI REVUE

Abo-Service 9
Postfach 1107
8044 UNTERSCHLEISSHEIM

RESERVIERUNGS-SERVICE

Assembler bietet dem, der damit umgehen kann, viele Möglichkeiten. Andererseits aber sind Assembler-Programme beim Abdruck im Heft sehr platzraubend. Aus diesem Grund erarbeitet die Redaktion gerade ein Assembler-Special für alle Freunde dieser Programmiersprache.

Dieses Heft wird voraussichtlich etwa zwischen dem 20. Dezember und 15. Januar 86 erscheinen, aber nur an ganz gezielt ausgewählten Verkaufsstellen angeboten — Bahnhofskiosken — Computershops usw.

Damit Sie nicht auf die Suche gehen müssen, können Sie sich dieses Heft schon jetzt reservieren lassen. Einfach den Coupon ausfüllen und — im Briefumschlag als Drucksache oder auf eine Postkarte geklebt — absenden.

Als Gegenleistung für diese Ihre Mühe erhalten Sie das Heft als Subskriptionsangebot sogar noch zehn Prozent billiger — einschließlich unserer Versandspesen. Nämlich für DM 17,80 statt zum Ladenpreis von DM 19,80.

ACHTUNG! Bitte kein Geld mitsenden, wir melden uns, wenn das Heft versandbereit ist.

Auf besonderen Wunsch versenden wir auch über Nachnahme. Wir weisen aber ausdrücklich darauf hin, daß dadurch Mehrkosten von etwa DM 5,— Mark im In- und etwa DM 10,— im Ausland (Postgebühren!) entstehen.

RESERVIERUNGS-KARTE

Bitte reservieren Sie mir . . . Exemplar(e) TI ASSEMBLER SPECIAL zum Vorzugspreis von DM 17,80.

Ich zahle:
- □ Nach Erhalt Ihrer Vorausrechnung (Versand am Tag des Geldeingangs).
- □ Per Nachnahme.
- □ Per Bankabbuchung am Versandtag.

Kto-Nr.: ________ Bankleitzahl ________

Bei (Bank und Ort) ________

Name ________

Straße/Hausnr. ________

Ort/PLZ ________

Unterschrift ________

PROGRAMMSERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkeufsbedingungen die Listings dieses Heftes euf

- □ Kessette (10 DM)
- □ Diskette (25 DM)

Zutreffendes bitte ankreuzen!

Ich zahle:
per beigefügtem Scheck / Schein ( )
Gegen Vorausrechnung ( ) — Versand em Tag des Geldeinganges
Gegen Bankabbuchung em Versandtag ( )

Meine Bank (mit Ortsname) . . . .

Meine Kontonummer . . . .

Meine Bankleitzahl . . . . (steht euf jedem Bankauszug)

Vorneme . . . . Nechname . . . .

Str./Nr. . . . . PLZ / Ort . . . .

Hiermit bestätige ich mit meiner Unterschrift, Ihre Verkaufsbedingungen gelesen zu haben und zu akzeptieren.

Unterschrift . . . .

Bitte ausschneiden und einsenden an
TI-REVUE
KASSETTENSERVICE 8
Postfach 1107
8044 Unterschleißheim

```
1600 SUB HIGH(H):: DISPLAY A
T(1,19)SIZE(7):H :: SUBEND
1610 SUB TIME(T):: T=T-1 ::
DISPLAY AT(1,1)SIZE(4):T ::
SUBEND
1620 SUB WAIT(B):: FOR J=1 T
O B :: NEXT J :: SUBEND
```

Fortsetzung von Seite 26

Joysticks in die gewünschte Richtung, kann ein gezielter Schuß ausgelöst werden. Der Gegner muß nun versuchen, entweder zuerst einen Treffer zu erzielen oder sich rechtzeitig aus der Schußbahn zu entfernen. Der Kaktus kann hierzu eine große Hilfestellung sein.
Ganz klar, daß für den zuerst getroffenen Spieler das Spiel verloren und damit zu Ende ist, denn tote Revolverhelden schießen nicht.

Anmerkung der Redaktion:
Manche verdammen zwar die Ballerspiele in den Spielhallen oder auf dem Heim-Computer. Dies ist jedoch kein Spiel, bei dem man alleine gegen den Computer antreten muß und der Kontaktarmut der Computer-Freaks weiteren Vorschub leistet, sondern dieses Spiel erfordert einen „menschlichen" Partner, der die andere Spielfigur steuert und durch Geschick und Einfallsreichtum versucht, die Situation zu meistern.
Aus diesen Gründen heraus sehen wir dieses Spiel mehr als einen Geschicklichkeitsvergleich zwischen zwei Partnern an und haben keine Bedenken, es zu veröffentlichen.



```
100 REM ******************
110 REM *  BILDMUSTER -    *
120 REM *   GENERATOR      *
130 REM * COPYRIGHT 1985  *
140 REM *Werner Liebender *
150 REM * bearbeitet vom  *
160 REM *     TEAM des     *
170 REM *AKTUELL VERLAGES *
180 REM *     MUENCHEN     *
190 REM *BENOTIGTE GERAETE*
200 REM * TI99/4A KONSOLE *
210 REM *KASSETTEN - REC. *
220 REM *                  *
230 REM * SPEICHERPLATZ    *
240 REM *    4816 BYTES    *
250 REM *                  *
260 REM ******************
270 CALL CLEAR
280 CALL SCREEN(2)
290 FOR X=1 TO 12
300 CALL COLOR(X,2,2)
310 NEXT X
320 PRINT TAB(5);"BILDMUSTER
GENERATOR"
330 PRINT TAB(5);"----------
---------": : :
340 PRINT ">1< VERTIKALE LIN
IEN": :
350 PRINT ">2< HORIZONTALE L
LINIEN": :
360 PRINT ">3< GITTERMUSTER"
: :
370 PRINT ">4< WEISSES RASTE
R": :
380 PRINT ">5< SCHWARZES RAS
TER": :
390 PRINT ">6< PUNKTRASTER":
 :
400 PRINT ">7< FARBBALKEN":
:
410 PRINT ">8< FARBEN": : :
420 PRINT "-----------------
-----------"
430 PRINT "...MIT LEERTASTE
ZURUECK"
440 CALL SCREEN(15)
450 FOR X=1 TO 12
460 CALL COLOR(X,2,15)
470 NEXT X
480 CALL KEY(0,K,S)
490 IF S=0 THEN 480
500 IF K<49 THEN 480
510 IF K>56 THEN 480
520 ON K-48 GOTO 530,630,730
,1040,1080,1120,1200,1610
530 CALL CLEAR
540 REM *VERTIKALE LINIEN*
550 CALL SCREEN(2)
560 CALL COLOR(1,16,2)
```

```
570 A$="1010101010101010"
580 CALL CHAR(35,A$)
590 FOR X=1 TO 32 STEP 2
600 CALL VCHAR(1,X,35,24)
610 NEXT X
620 GOTO 2130
630 CALL CLEAR
640 REM *HORIZONTALE LINIEN*
650 CALL SCREEN(2)
660 CALL COLOR(1,16,2)
670 B$="000000FF00000000"
680 CALL CHAR(36,B$)
690 FOR X=1 TO 24 STEP 2
700 CALL HCHAR(X,1,36,32)
710 NEXT X
720 GOTO 2130
730 CALL CLEAR
740 REM *GITTERMUSTER*
750 CALL SCREEN(2)
760 CALL COLOR(1,16,2)
770 A$="1010101010101010"
780 B$="000000FF00000000"
790 C$="101010FF10101010"
800 CALL CHAR(35,C$)
810 CALL CHAR(36,A$)
820 CALL CHAR(37,B$)
830 FOR X=1 TO 32 STEP 2
840 FOR Z=1 TO 24 STEP 2
850 CALL VCHAR(Z,X,35,1)
860 NEXT Z
870 NEXT X
880 FOR X=1 TO 32 STEP 2
890 FOR Z=2 TO 24 STEP 2
900 CALL VCHAR(Z,X,36,1)
910 NEXT Z
920 NEXT X
930 FOR X=2 TO 32 STEP 2
940 FOR Z=1 TO 24 STEP 2
950 CALL VCHAR(Z,X,37,1)
960 NEXT Z
970 NEXT X
980 CALL COLOR(1,16,2)
990 GOTO 2130
1000 CALL SCREEN(2)
1010 CALL COLOR(1,2,2)
1020 CALL COLOR(1,16,2)
1030 GOTO 2130
1040 CALL CLEAR
1050 CALL SCREEN(16)
1060 CALL COLOR(1,16,16)
1070 GOTO 2130
1080 CALL CLEAR
1090 CALL SCREEN(2)
1100 CALL COLOR(1,2,2)
1110 GOTO 2130
1120 CALL CLEAR
1130 REM *PUNKTRASTER*
1140 CALL SCREEN(2)
1150 CALL COLOR(1,16,2)
```

```
1160 C$="0000000010000000"
1170 CALL CHAR(37,C$)
1180 CALL VCHAR(1,1,37,864)
1190 GOTO 2130
1200 CALL CLEAR
1210 REM *FARBBALKEN*
1220 CALL SCREEN(2)
1230 D$="FFFFFFFFFFFFFFFF"
1240 CALL COLOR(1,7,2)
1250 CALL CHAR(33,D$)
1260 CALL VCHAR(1,5,33,48)
1270 CALL COLOR(2,9,15)
1280 CALL CHAR(41,D$)
1290 CALL VCHAR(1,7,41,48)
1300 CALL COLOR(3,10,15)
1310 CALL CHAR(48,D$)
1320 CALL VCHAR(1,9,48,48)
1330 CALL COLOR(4,13,15)
1340 CALL CHAR(58,D$)
1350 CALL VCHAR(1,11,58,48)
1360 CALL COLOR(5,3,15)
1370 CALL CHAR(64,D$)
1380 CALL VCHAR(1,13,64,48)
1390 CALL COLOR(6,4,15)
1400 CALL CHAR(74,D$)
1410 CALL VCHAR(1,15,74,48)
1420 CALL COLOR(7,11,15)
1430 CALL CHAR(81,D$)
1440 CALL VCHAR(1,17,81,48)
1450 CALL COLOR(8,12,15)
1460 CALL CHAR(92,D$)
1470 CALL VCHAR(1,19,92,48)
1480 CALL COLOR(9,14,15)
1490 CALL CHAR(96,D$)
1500 CALL VCHAR(1,21,96,48)
1510 CALL COLOR(10,5,15)
1520 CALL CHAR(104,D$)
1530 CALL VCHAR(1,23,104,48)
1540 CALL COLOR(11,6,15)
1550 CALL CHAR(112,D$)
1560 CALL VCHAR(1,25,112,48)
1570 CALL COLOR(12,8,15)
1580 CALL CHAR(123,D$)
1590 CALL VCHAR(1,27,123,48)
1600 GOTO 2130
1610 CALL CLEAR
1620 REM  *FARBEN*
1630 CALL SCREEN(2)
1640 FOR X=1 TO 12
1650 CALL COLOR(X,2,2)
1660 NEXT X
1670 PRINT TAB(9);"FARBAUSWA
HL"
1680 PRINT TAB(9);"---------
--": : :
1690 PRINT ">1<  ROT"
1700 PRINT ">2<  GRUEN"
1710 PRINT ">3<  BLAU"
1720 PRINT ">4<  GELB"
```

```
1730 PRINT ">5<  MAGENTA"
1740 PRINT ">6<  GRAU": : :
: : : : :
1750 CALL SCREEN(15)
1760 FOR X=1 TO 12
1770 CALL COLOR(X,2,15)
1780 NEXT X
1790 CALL KEY(0,K,S)
1800 IF S=0 THEN 1790
1810 IF K=32 THEN 270
1820 IF K>56 THEN 1790
1830 IF K<49 THEN 1790
1840 ON K-48 GOTO 1850,1890,
1930,1970,2010,2050
1850 CALL CLEAR
1860 CALL SCREEN(9)
1870 CALL COLOR(1,9,9)
1880 GOTO 2090
1890 CALL CLEAR
1900 CALL SCREEN(3)
1910 CALL COLOR(1,3,3)
1920 GOTO 2090
1930 CALL CLEAR
1940 CALL SCREEN(6)
1950 CALL COLOR(1,6,6)
1960 GOTO 2090
1970 CALL CLEAR
1980 CALL SCREEN(11)
1990 CALL COLOR(1,11,11)
2000 GOTO 2090
2010 CALL CLEAR
2020 CALL SCREEN(14)
2030 CALL COLOR(1,14,14)
2040 GOTO 2090
2050 CALL CLEAR
2060 CALL SCREEN(15)
2070 CALL COLOR(1,15,15)
2080 GOTO 2090
2090 CALL KEY(0,K,S)
2100 IF S=0 THEN 2090
2110 IF K<>32 THEN 2090
2120 IF K=32 THEN 1610
2130 CALL KEY(0,K,S)
2140 IF S=0 THEN 2130
2150 IF K<>32 THEN 2130
2160 IF K=32 THEN 270
```

BÖRSE


```
100 ! *********************
110 ! *                   *
120 ! *   W E S T E R N   *
130 ! *                   *
140 ! *    Copyright by   *
150 ! *                   *
160 ! *    H.-P. KREUTZ   *
170 ! *                   *
190 ! * Benoetigte Geraete *
200 ! *  TI99/4A  Konsole *
210 ! *     Ext. Basic    *
220 ! *                   *
260 ! *  Speicherbelegung *
270 ! *     4470 Bytes    *
280 ! *                   *
290 ! *********************
300 CALL CLEAR :: CALL SCREE
N(12)
310 CALL CHAR(91,"3C4299A1A1
99423C",104,"1858587A1A1E181
8")
320 CALL CHAR(105,"1858587A1
A1E182E",106,"24A495492A1C18
3E")
330 CALL CHAR(100,"00000000
0000003030000000000000000000
00000000C0C00000000000000000"
)
340 CALL CHAR(108,"01010709
10003672B11000102040400CC0C8F
0C0C080E0F3EAC400C020101018"
)
350 CALL CHAR(112,"02010001
10003672B1100010204040C20408
0C0C080E0F3EAC400C020101018"
)
360 CALL CHAR(40,"000000000
0000000C8780F50D9DA047000000
0000000000000080484DEFEDE4482")
370 CALL CHAR(60,"000000000
0000003804825B3FDB0201000000
0000000000000091562607905E2")
380 CALL CHAR(132,"01042000 6
844020101000000000044000C0460
084808CAEC6C0A690819404401 0"
)
390 CALL CHAR(36,"0040201C1C
1D03070F17234020100000000C08
10E00080C0C0F9858280F01020")
:: CALL CHAR(119,"8484A4A5ED
EDFFFF")
400 CALL CHAR(96,"0000020508
1810000003242810000000020160F
0F8F7E797C7AF121212110 0818")
410 RANDOMIZE :: P1,P2=0 ::
S1=1 :: S2=0 :: A1,A2=0
420 CALL CLEAR :: CALL MAGNI
FY(3)
430 DISPLAY AT(6,8):"W E S T
```

```
 E R N" :: CALL SPRITE(#1,1Ø
8,5,8Ø,198,Ø,Ø):: CALL SPRIT
E(#2,112,7,8Ø,4Ø,Ø,Ø)
44Ø CALL COLOR(11,13,1):: CA
LL HCHAR(12,1,119,32):: DISP
LAY AT(15,4):"PRESS ANY KEY
TO START"
45Ø DISPLAY AT(22,6):"I KREU
TZ,HANS-PETER" :: DISPLAY AT
(24,8):"2ØØØ HAMBURG 71"
46Ø CALL SOUND(5ØØ,146,Ø)::
CALL SOUND(25Ø,13Ø,Ø):: CALL
 SOUND(25Ø,123,Ø):: CALL SOU
ND(125,146,Ø):: CALL SOUND(2
5Ø,195,Ø)
47Ø CALL SOUND(125,22Ø,Ø)::
CALL SOUND(1ØØØ,246,Ø):: CAL
L SOUND(5ØØ,195,Ø):: CALL SO
UND(2ØØ,3ØØØ,3Ø):: CALL SOUN
D(1ØØØ,196,Ø,262,Ø,33Ø,Ø)
48Ø FOR I=1 TO 4Ø :: NEXT I
49Ø CALL KEY(Ø,K,S):: IF S=Ø
 THEN 49Ø
5ØØ P1,P2=Ø :: CALL CLEAR ::
 CALL SCREEN(12)
51Ø CALL HCHAR(1,1,119,32)::
 DISPLAY AT(23,2):"INDIAN 1:
.........COWBOY 2:"
52Ø DISPLAY AT(24,3):"";P1 :
: DISPLAY AT(24,21):"";P2
53Ø CALL COLOR(1Ø,13,15)
54Ø FOR I=1 TO 12
55Ø X=INT(RND*2Ø)+1 :: Y=INT
(RND*28)+1 :: CALL HCHAR(X+2
,Y,1Ø5):: CALL HCHAR(X+1,Y,1
Ø4)
56Ø N=INT(RND*27)+1 :: M=INT
(RND*27)+1
57Ø DISPLAY AT(N+2,M)SIZE(1)
:CHR$(1Ø6)
58Ø NEXT I
59Ø CALL SPRITE(#1,112,9,(RN
D*2ØØ)+1,32)
6ØØ CALL SPRITE(#2,1Ø8,5,(RN
D*2ØØ)+1,24Ø)
61Ø CALL JOYST(1,X1,Y1):: IF
 X1<>Ø THEN 66Ø :: CALL MOTI
ON(#1,-2*Y1,Ø)
62Ø CALL JOYST(2,X2,Y2):: IF
 X2<>Ø THEN 69Ø :: CALL MOTI
ON(#2,-2*Y2,Ø)
63Ø CALL KEY(1,K,S):: IF K=1
8 THEN 72Ø
64Ø CALL KEY(2,K,S):: IF K=1
8 THEN 85Ø
65Ø GOTO 61Ø
66Ø IF X1=4 THEN 67Ø ELSE 68
Ø
67Ø S1=1 :: CALL PATTERN(#1,
112):: CALL MOTION(#1,Ø,8)::
 GOTO 62Ø
68Ø S1=Ø :: CALL PATTERN(#2,
1Ø8):: CALL MOTION(#1,Ø,-8):
: GOTO 62Ø
69Ø IF X2=4 THEN 7ØØ ELSE 71
Ø
7ØØ S2=1 :: CALL PATTERN(#1,
112):: CALL MOTION(#2,Ø,8)::
 GOTO 63Ø
71Ø S2=Ø :: CALL PATTERN(#2,
1Ø8):: CALL MOTION(#2,Ø,-8):
: GOTO 63Ø
72Ø CALL MOTION(#1,Ø,Ø):: CA
LL POSITION(#1,A,B)
73Ø IF S1=1 THEN 75Ø
74Ø CALL SOUND(5Ø,-6,Ø):: CA
LL SOUND(7Ø,-7,Ø):: CALL SPR
ITE(#8,132,16,A-1,B-8,Ø,Ø)::
 CALL SPRITE(#3,1ØØ,16,A-1,B
-8,Ø,-4Ø):: GOTO 76Ø
75Ø CALL SOUND(5Ø,-6,Ø):: CA
LL SOUND(8Ø,-7,Ø):: CALL SPR
ITE(#7,132,16,A-1,B+8,Ø,Ø)::
 CALL SPRITE(#3,1ØØ,16,A-1,B
+8,Ø,4Ø)
76Ø CALL DELSPRITE(#7,#8)
77Ø FOR I=1 TO 15
78Ø CALL COINC(ALL,C):: IF C
 THEN 82Ø
79Ø CALL JOYST(2,X2,Y2):: CA
LL MOTION(#2,-2*Y2,2*X2)
8ØØ NEXT I
81Ø CALL DELSPRITE(#3):: P2=
P2+2Ø :: DISPLAY AT(24,21):"
";P2 :: GOTO 61Ø
82Ø CALL DELSPRITE(#3):: CAL
L MOTION(#2,Ø,Ø):: FOR I=1 T
O 4Ø :: CALL PATTERN(#2,96):
: NEXT I :: CALL PATTERN(#2,
4Ø)
83Ø P1=P1+1ØØ :: DISPLAY AT(
24,3)SIZE(1Ø):"";P1
84Ø FOR I=1 TO 2ØØ :: NEXT I
 :: CALL PATTERN(#2,1Ø8):: A
1=A1+1 :: IF A1=5Ø THEN
98Ø :: GOTO 61Ø
85Ø CALL MOTION(#2,Ø,Ø):: CA
LL POSITION(#2,A,B)
86Ø IF S2=1 THEN 88Ø
87Ø CALL SOUND(5Ø,-6,Ø):: CA
LL SOUND(7Ø,-7,Ø):: CALL SPR
ITE(#8,132,16,A-1,B-8,Ø,Ø)::
 CALL SPRITE(#4,1ØØ,16,A-1,B
-8,Ø,-4Ø):: GOTO 89Ø
88Ø CALL SOUND(5Ø,-6,Ø):: CA
LL SOUND(7Ø,-7,Ø):: CALL SPR
ITE(#7,132,16,A-1,B+8,Ø,Ø)::
 CALL SPRITE(#4,1ØØ,16,A-1,B
+8,Ø,4Ø)
89Ø CALL DELSPRITE(#7,#8)
9ØØ FOR I=1 TO 15
91Ø CALL COINC(ALL,C):: IF C
```

```
 THEN 950
920 CALL JOYST(1,X1,Y1):: CA
LL MOTION(#1,-2*Y1,2*X1)
930 NEXT I
940 CALL DELSPRITE(#4):: P1=
P1+20 :: DISPLAY AT(24,3)SIZ
E(10):"";P1 :: GOTO 610
950 CALL DELSPRITE(#4):: CAL
L MOTION(#1,0,0):: FOR I=1 T
O 40 :: CALL PATTERN(#1,36):
: NEXT I :: CALL PATTERN(#1,
60)
960 P2=P2+100 :: DISPLAY AT(
24,21):"";P2
970 FOR I=1 TO 200 :: NEXT I
 :: CALL PATTERN(#1,112):: A
2=A2+1 :: IF A1=50 THEN
980 :: IF A2=50 THEN 980 ::
GOTO 610
980 FOR I=1 TO 3 :: CALL SOU
ND(100,262,0):: CALL SOUND(1
00,330,0):: CALL SOUND(100,3
92,0):: NEXT I
990 DISPLAY AT(10,6):"G A M
E - O V E R" :: FOR I=1 TO 1
00 :: NEXT I
1000 DISPLAY AT(10,6):"" ::
FOR I=1 TO 100 :: NEXT I
1010 DISPLAY AT(24,2):"WISH
YOU A NEW GAME...  J/N" :: C
ALL KEY(0,K,S):: IF S=0 THEN
 990 :: IF K=78 THEN 1030 EL
SE 1020
1020 CALL DELSPRITE(ALL):: G
OTO 410
1030 END
```

KAVALKADE

Die Idee dazu hatte ich beim Reversi spielen. Der Grundgedanke ist folgender: Zwei Spieler setzen abwechselnd Steine auf die Spielfelder und können sie sich gegenseitig abjagen. Auf jedes Feld können mehrere Steine gesetzt werden. Sind aber auf einem Feld so viele Steine wie Nachbarfelder, so werden diese auf alle benachbarten Felder verteilt. Nur so kann man gegnerische Steine für sich gewinnen. Wenn einer der Steine auf ein vom Gegner besetztes Feld fällt, werden dann die fremden Steine zu eigenen.
Gewonnen hat, wer seinem Mitspieler alle Spielsteine abgenommen hat. Hört sich nicht besonders kompliziert an. Die auftretenden Kettenreaktionen konnte ich aber nur durch zwei Unterprogramme, die sich gegenseitig mit GOSUB aufrufen, in den Griff bekommen. Jede andere Lösung wäre viel zu langsam. Um das Programm weiter zu beschleunigen, mußte X-Basic her. Es ginge natürlich auch in Basic. Aber gegen Spielende sind die Wartezeiten auch so recht lang. Noch schneller wäre KAVALKADE, wenn man die „Begleitmusik" wegläßt und die neue Spielsituation erst auf dem Bildschirm darstellt, wenn die Berechnung abgeschlossen ist. Aber das würde recht langweilig, weil sich auf dem Fernseher dann nicht mehr viel tut.
Übrigens läßt sich das Spiel leicht verändern, z.B. indem man das Prüffeld anders belegt.
Und: Keine Sorge, der Stapel mit den Rücksprungadressen kann nicht größer als 260 Byte werden. Wenn nämlich 261 Steine auf dem Brett sind, gibt es eine „ewige" Kettenreaktion. Sobald aber alle Steine eine Farbe haben, bricht das Spiel ab. Einer von beiden Spielern hat gewonnen. *Reinhardt Kern*

TIEFKÜHLKOST

Haben Sie Ihre Tiefkühltruhe voll bis oben hin, wissen Sie noch was Sie alles gelagert haben und wann die einzelnen Lebensmittel verfallen? Nein?
Ab sofort ist dies kein Problem mehr für Sie. Lassen Sie doch den Computer den Inhalt Ihrer Truhe verwalten.
In Sekundenschnelle wissen Sie, welche Lebensmittel überlagert sind oder sofort verbraucht werden müssen.
Ein komfortables Menü führt Sie durch das ganze Programm. Als Auswahlmöglichkeiten stehen Ihnen Kassettenrekorder oder Diskettenstation, sowie 10 weitere Funktionen zur Verfügung. Wollen Sie die einzelnen Funktionen vor Programmbeginn erklärt haben, so rufen Sie durch Drücken der Taste -J- die Beschreibung auf.
Den Inhalt Ihrer Tiefkühltruhe können Sie in 5 Kategorien trennen:
1. Fleisch, 2. Gemüse,
3. Fisch, 4. Backwaren,
5. Sonstiges.
Ist die Kategorie gewählt, fordert der Computer nun Angaben über das Lebensmittel, welches Sie einfrieren wollen. Im Dialog mit dem Computer geben Sie nun Art des Lebensmittels, Stückzahl, Gewicht, Haltbarkeitsdatum und Tag des Einfrierens ein. In der letzten Zeile haben Sie noch die Möglichkeit eine Bemerkung zu dem Lebensmittel abzugeben. Sollten Ihnen die Angaben zu umfangreich sein, drücken Sie bei den Angaben, die Sie nicht für nötig halten einfach -ENTER- Lebensmittel, die aus der Gefriertruhe genommen werden, können schnell und problemlos gelöscht werden.
Wollen Sie eine Gesamtübersicht über den Inhalt, würden Sie gerne wissen, welche Fleischsorten noch vorrätig sind oder ob Sie das eingefrorene Gemüse nächsten Sonntag noch verwenden können? Fragen Sie den Computer, er hat es für Sie gespeichert. Selbstverständlich können Sie den Inhalt Ihrer Truhe auch sortieren lassen. Möchten Sie das Programm beenden, so geschieht dies mit der -9- im Hauptmenü. Zuvor speichert der Computer die einzelnen Datensätze jedoch wieder auf das gewählte Speichersystem zurück, damit Ihnen keine Informationen verloren gehen.
Haben Sie noch keine Tiefkühltruhe? Nachdem Sie dieses Programm ausgetestet haben, werden Sie sich bestimmt bald eine zulegen.
B. Knedel

Die nächste TI-REVUE am 27. Dez.

```
10 ! **********************
11 ! *                    *
12 ! *     KAVALKADE      *
13 ! * ein Strategiespiel *
14 ! *  fuer 2 Personen   *
15 ! *    Copyright by    *
16 ! *   Reinhard Kern    *
17 ! *                    *
18 ! *                    *
19 ! * Benoetigte Geraete *
2Ø ! *  TI99/4A  Konsole  *
21 ! *     Ext. Basic     *
22 ! *                    *
26 ! *  Speicherbelegung  *
27 ! *     7756 Bytes     *
28 ! *                    *
29 ! **********************
3Ø ! Speicherbedarf nach
  Dimensionierung der Felder
  und ohne Kopfbild.
2ØØ DIM SF1(9,9),SF2(9,9),PF
(9,9)
21Ø !** PGMABTAST ABSCHALTEN
22Ø GOTO 27Ø !VARIABLENLISTE
:
23Ø C,CT,I,J,Z(1),Z(2),B$,SP
,SN,X,Y,K,S,E,F
24Ø !** CALL-LISTE **
25Ø CALL CLEAR :: CALL COLOR
 :: CALL SCREEN :: CALL CHAR
 :: CALL CHARPAT :: CALL SOU
ND :: CALL HCHAR :: CALL KEY
26Ø CALL SPRITE :: CALL LOCA
TE
27Ø !@P-
28Ø CALL CHAR(123,"ØØ44ØØ384
47C4444ØØ44ØØ7C4444447CØØ44Ø
Ø444444438")!UMLAUTE AE,OE,
UE(KLEIN)
29Ø CT=2^(1/12)!HALBTONOPERA
TOR
3ØØ !*** INTRO ***
31Ø CALL COLOR(2,1,1):: CALL
 CLEAR :: CALL SCREEN(5)
32Ø FOR I=1 TO 3 :: PRINT "*
*..*";TAB(25);"**" :: NEXT I
 :: PRINT "**.*...***..**..*
..***..**"
33Ø PRINT "***...**..*.**..*
.**..*.**..****..**..*.**..*
.**..*.**..**.**.*****.*****
.*****.**..**..*.**..*..***.
.**..*.**"
34Ø PRINT "**..*.**..*...*..
.**..*.****": :RPT$("*",28):
 :
35Ø FOR I=1 TO 3 :: PRINT ".
....*";TAB(22);"*" :: NEXT I
36Ø PRINT ".....*..**..***..
.**.*..***......*.**..*****.
**.**.***.*.....***...**..*.
**..*.**..*.....**.*..**..*.
**..*.**.*"
37Ø PRINT ".....**.*..*****
**..*.***......**..*.**..*.
**..*.**..*.....**..*.**..*.
.****..***.....SPIELANLEITUN
G? <J\N>"
38Ø CALL COLOR(2,RND*12+4,3)
:: CALL SOUND(-4ØØ,11Ø*CT^(R
ND*48+1),2):: CALL KEY(3,K,S
):: IF S=Ø THEN 38Ø
39Ø IF K=78 THEN GOTO 5ØØ EL
SE IF K<>74 THEN GOTO 38Ø
4ØØ !*** SPIELREGELN ***
41Ø CALL CLEAR :: FOR I=Ø TO
 12 :: CALL COLOR(I,1,1):: N
EXT I :: CALL SCREEN(4)
42Ø PRINT TAB(1Ø);"KAVALKADE
":"ein Strategiespiel f)r Zw
ei": :"Staple Steine auf dei
ne Fel-der, bis eines explod
iert. "
43Ø PRINT "Die Tr)mmer werde
n auf die..Nachbarfelder ver
teilt und..k!nnen Felder des
 Gegners ineigene umwandeln.
": :
44Ø PRINT "Starte die Ketten
reaktion...und besiege Deine
n Gegner!": :"Beeile Dich! D
ein Gegner....kann Dich im n
{chsten Zug...besiegen!": :
45Ø PRINT "Steuere mit den P
feiltasten und setze Steine
mit 'A'. ..Aber: Falsch gese
tzte Steinek!nnen nicht zur)
ckgesetzt..werden!": :
46Ø PRINT "Bitte eine Taste
dr)cken.";
48Ø FOR I=Ø TO 12 :: CALL CO
LOR(I,16,5):: NEXT I
49Ø CALL KEY(Ø,K,S):: IF S=Ø
 THEN 49Ø
5ØØ !*** SPIELFELD ERSTELLEN
51Ø CALL CLEAR :: FOR I=Ø TO
 12 :: CALL COLOR(I,2,1):: N
EXT I
52Ø CALL CHARPAT(63,B$):: CA
LL CHAR(64,B$)
53Ø CALL CHAR(126,""):: B$="
ØØØØ183C3C18ØØØØØØ3C7E7E7E7E
3CØØ3C7EFFE7E7FF7E3CFFFFFFFF
FFFFFFFF" :: CALL CHAR(6Ø,B$
):: CALL CHAR(1Ø8,B$)
54Ø CALL COLOR(3,16,4,4,16,4
,9,13,8,1Ø,13,8,12,5,16)
55Ø CALL SCREEN(11)
56Ø FOR I=5 TO 8 :: CALL COL
OR(I,13,1):: NEXT I
```

```
570 FOR I=48 TO 57 :: CALL C
HARPAT(I,B$):: CALL CHAR(I+5
Ø,B$):: NEXT I !ZWEITEN ZAHL
ENBEREICH EINRICHTEN(98-107)
580 FOR I=1 TO 8 :: PF(I,Ø),
PF(Ø,I)=3 :: FOR J=1 TO 8 ::
 PF(I,J)=4 :: NEXT J :: PF(I
,9),PF(9,I)=3 :: NEXT I
590 PF(Ø,Ø),PF(Ø,9),PF(9,Ø),
PF(9,9)=2 !PRUEFFELD BELEGT
600 FOR I=3 TO 12 :: CALL HC
HAR(I,7,126,10):: NEXT I !SP
IELBRETT
610 DISPLAY AT(2,5):"ABCDEFG
HIJ" :: DISPLAY AT(13,5):"AB
CDEFGHIJ"
620 FOR I=1 TO 10 :: CALL HC
HAR(2+I,6,47+I,1):: CALL HCH
AR(2+I,17,97+I,1):: NEXT I
630 CALL CHAR(97,"FF81818181
8181FF"):: CALL SPRITE(#2,97
,2,17,49)
640 DISPLAY AT(15,1): : : :
: : :
650 FOR I=Ø TO 9 :: FOR J=Ø
TO 9 :: SF1(I,J),SF2(I,J)=Ø
:: NEXT J :: NEXT I
700 !**** HAUPTPROGRAMM
710 !** STEINE SETZEN
720 Z(1),Z(2),X,Y=Ø !ZAEHLER
,CURSORPOSITION
730 SP=2 !SP=SPIELER AM ZUG
740 SN=1 !SN=NICHT AM ZUG
750 SP=-1*(SP=2)-2*(SP=1)
760 SN=-1*(SN=2)-2*(SN=1)
770 DISPLAY AT(18,2):"AM ZUG
 IST SPIELER";SP
780 DISPLAY AT(20,2):"SETZE
EINEN STEIN."
790 CALL SOUND(-200,880,4,-8
,Ø)
800 !* EINGABE PER CURSOR
820 CALL KEY(1,K,S):: K=K+1
:: IF K=2 THEN GOTO 910 ELSE
 IF S=Ø THEN GOTO 820
830 E=(K=3)+(K=5)+(K=16)-(K=
4)-(K=7)-(K=15):: F=(K=1)+(K
=16)+(K=15)-(K=6)-(K=5)-(K=7
)
840 X=X+E :: Y=Y-F
850 X=INT(10*(X/10-INT(X/10)
))
860 Y=INT(10*(Y/10-INT(Y/10)
))
870 CALL LOCATE(#2,1+(Y+2)*8
,1+(X+6)*8):: GOTO 820
900 !AUSWERTEN DER EINGABE
910 IF SF2(X,Y)=SN THEN CALL
 SOUND(200,-5,Ø,880,Ø):: GOT
O 770
920 SF2(X,Y)=SP :: SF1(X,Y)=
SF1(X,Y)+1 :: Z(SP)=Z(SP)+1
!Z=ANZAHL DER STEINE,SF ERHO
EHEN
930 DISPLAY AT(5,20):"Z(1)";
Z(1):: DISPLAY AT(7,20):"Z(d
)";Z(2)
940 CALL HCHAR(Y+3,X+7,SF1(X
,Y)+59-48*(SP=2))!DARSTELLEN
 AM BILDSCHIRM
950 IF SF1(X,Y)>=PF(X,Y)THEN
 GOSUB 1000 :: C=Ø
960 GOTO 750
1000 !*** EXPLODIEREN ***
1010 !ALTEN STEIN LOESCHEN
1020 SF1(X,Y),SF2(X,Y)=Ø ::
CALL HCHAR(Y+3,X+7,126)
1030 !* NEUE STEINE SETZEN *
1040 IF Y>Ø THEN Y=Y-1 :: GO
SUB 1100 :: Y=Y+1
1050 IF X>Ø THEN X=X-1 :: GO
SUB 1100 :: X=X+1
1060 IF X<9 THEN X=X+1 :: GO
SUB 1100 :: X=X-1
1070 IF Y<9 THEN Y=Y+1 :: GO
SUB 1100 :: Y=Y-1
1090 RETURN
1100 !** STEINE SETZEN **
1120 Z(SN)=Z(SN)+SF1(X,Y)*(S
F2(X,Y)=SN)
1130 Z(SP)=Z(SP)-SF1(X,Y)*(S
F2(X,Y)=SN)
1140 SF2(X,Y)=SP
1150 SF1(X,Y)=SF1(X,Y)+1 !ST
EINE REGULIERT
1160 DISPLAY AT(5,20):"Z(1)"
;Z(1):: DISPLAY AT(7,20):"Z(
d)";Z(2)
1170 CALL SOUND(-450,440*CT^
(X+Y+1),1)
1180 CALL HCHAR(Y+3,X+7,SF1(
X,Y)+59-48*(SP=2))
1190 IF Z(SN)<1 THEN GOTO 12
50
1200 IF SF1(X,Y)>=PF(X,Y)THE
N GOSUB 1000
1220 RETURN
1230 !*** SPIELENDE ***
1250 FOR I=1 TO 3 :: CALL SO
UND(200,110,1,220,1):: CALL
SOUND(900,349,Ø,440,2,523,3,
-8,5):: NEXT I
1270 DISPLAY AT(15,1): : : :
 : : : : : :: DISPLAY AT(16,
1):"BRAVO, SPIELER";SP:"HAT
GEWONNEN.": :"WOLLT IHR NOCH
MAL SPIELEN@";"(J/N)"
1280 CALL KEY(3,K,S):: IF S=
Ø THEN GOTO 1280
1290 IF K=74 THEN GOTO 600 E
LSE IF K<>78 THEN 1280
1300 RUN "" !=NEW
```

```
100 ! **********************
110 ! *                    *
120 ! *   Tiefkuehlkost    *
130 ! *   Copyright by     *
140 ! *     B. Knedel      *
150 ! *                    *
160 ! *  Bearbeitet vom    *
170 ! *     Team des       *
180 ! * Aktuell Verlages   *
190 ! *     Muenchen       *
200 ! *                    *
210 ! * Benoetigte Geraete *
220 ! * TI99/4A  Konsole   *
230 ! *     Ext. Basic     *
240 ! * Kass.Rec od. Floppy*
250 ! * Drucker moeglich   *
260 ! *                    *
270 ! *  Speicherbelegung  *
280 ! *     8964 Bytes     *
290 ! **********************
300 !
310 ON WARNING NEXT
320 OPTION BASE 1 :: CALL CL
EAR :: PRINT ".......TIEFKUE
HLKOST": : : : : : : : :"DIS
KETTE ODER CASSETTE?": :"D O
DER C": :
330 CALL KEY(0,K,S):: IF S=0
 THEN 330
340 I$="DSK1.TIEF"
350 IF K=68 OR K=100 THEN 37
0 ELSE IF K=67 OR K=99 THEN
360 ELSE 330
360 I$="CS1"
370 DIM NR(50),KA$(50),A$(50
),ST(50),GR(50),T(50),M(50),
J(50),BE$(50),D$(50)
380 PRINT "BITTE DATUM EINGE
BEN.": : :
390 INPUT "TAG....TT? ":T2 :
: INPUT "MONAT..MM? ":M2 ::
INPUT "JAHR...JJ? ":J2
400 PRINT : : : : : : : : :"
BESCHREIBUNG (J/N)"
410 CALL KEY(0,K,S):: IF S=0
 THEN 410
420 IF K=74 OR K=106 THEN 26
10 ELSE IF ((K<>78)OR(K<>110
))=0 THEN 410
430 CALL CLEAR
440 PRINT "....PROGRAMMENUE"
: : : :
450 PRINT "....1 NEUEINGABE-
ERGAENZEN"
460 PRINT "....2 AENDERN"
470 PRINT "....3 EINZELLOESC
HUNG"
480 PRINT "....4 KATEGORIESU
CHE"
490 PRINT "....5 EINZELSUCHE
"
500 PRINT "....6 GESAMTINHAL
T"
510 PRINT "....7 STICHTAGSUC
HE"
520 PRINT "....8 SORTIEREN"
530 PRINT "....9 PROGRAMMEND
E"
540 PRINT "...10 DATEN EINLE
SEN": : : :
550 INPUT "SIE WUENSCHEN? ":
CH
560 IF CH=9 THEN GOSUB 660 :
: END
570 IF CH=10 THEN 610
580 IF CH<1 OR CH>10 THEN 55
0
590 ON CH GOSUB 1660,1260,18
80,1000,2120,700,2270,2780
600 GOTO 430
610 OPEN #1:I$,INPUT ,FIXED
192,INTERNAL
620 INPUT #1:NF
630 FOR X=1 TO NF
640 INPUT #1:NR(X),KA$(X),A$
(X),ST(X),GR(X),T(X),M(X),J(
X),D$(X),BE$(X):: NEXT X
650 CLOSE #1 :: GOTO 430
660 OPEN #1:I$,OUTPUT,INTERN
AL,FIXED 192
670 PRINT #1:NF
680 FOR X=1 TO NF
690 PRINT #1:NR(X),KA$(X),A$
(X),ST(X),GR(X),T(X),M(X),J(
X),D$(X),BE$(X):: NEXT X ::
CLOSE #1 :: RETURN
700 CALL CLEAR :: PRINT "SIE
 HABEN";NF :: PRINT "LEBENSM
ITTEL GESPEICHERT." :: PRINT
 "LISTE AUF BILDSCHIRM ODER"
 :: PRINT "DRUCKER?"
710 PRINT : :"EINGABE=B ODER
 D" :: PRINT "ENDE=E"
720 CALL KEY(0,K,S):: IF S=0
 THEN 720
730 IF K=68 OR K=100 THEN 90
0
740 IF K=66 OR K=98 THEN 760
750 IF K=69 OR K=101 THEN 43
0 ELSE 720
760 CALL CLEAR :: FOR X=1 TO
 NF
770 PRINT "KATEGORIE=";KA$(X
):: PRINT "NR. :";NR(X)
780 PRINT "ART : ";A$(X):: P
RINT "STK.:";ST(X)
790 PRINT "GEW.:";GR(X)
800 GOSUB 2770
810 PRINT "HALTBAR BIS...: "
;X$
820 PRINT "EINGEFROREN AM: "
;D$(X)
```

```
830 PRINT "BEMERKUNG:"
840 PRINT BE$(X)
850 PRINT "WEITER=LEERTASTE"
 :: PRINT "ENDE=E"
860 CALL KEY(0,K,S):: IF S=0
 THEN 860
870 IF K=69 OR K=101 THEN 43
0
880 IF K<>32 THEN 860
890 CALL CLEAR :: NEXT X ::
RETURN
900 OPEN #2:"PIO"
910 PRINT #2:"LEBENSMITTELLI
STE..STAND : ";T2;".";M2;"."
;J2
920 FOR X=1 TO NF
930 PRINT #2:CHR$(14);
940 PRINT #2:KA$(X),A$(X)
950 PRINT #2:CHR$(15);
960 GOSUB 2770
970 PRINT #2:"NR.=";X,"GRAMM
=";GR(X),"STCK.=";ST(X),,"HA
LTBAR BIS: ";X$
980 PRINT #2:"BEMERKUNG:";BE
$(X)
990 NEXT X :: CLOSE #2 :: RE
TURN
1000 PRINT "KATEGORIE:"
1010 PRINT "1=FLEISCH" :: PR
INT "2=GEMUESE" :: PRINT "3=
FISCH" :: PRINT "4=BACKWAREN
" :: PRINT "5=SONSTIGES"
1020 PRINT :"SIE WUENSCHEN ?
 "
1030 CALL KEY(0,AA,S):: IF S
=0 THEN 1030
1040 IF AA<49 OR AA>53 THEN
1020
1050 AA=AA-48
1060 IF AA=1 THEN N$="FLEISC
H"
1070 IF AA=2 THEN N$="GEMUES
E"
1080 IF AA=3 THEN N$="FISCH"
1090 IF AA=4 THEN N$="BACKWA
REN"
1100 IF AA=5 THEN N$="SONSTI
GES"
1110 FOR X=1 TO NF
1120 IF KA$(X)<>N$ THEN 1250
1130 CALL CLEAR :: PRINT "KA
TEGORIE=";KA$(X):: PRINT "NR
....=";NR(X)
1140 PRINT "ART...= ";A$(X):
: PRINT "STK...=";ST(X)
1150 PRINT "GRAMM.=";GR(X)
1160 PRINT "HALTBARKEIT BIS
:";
1170 GOSUB 2770 :: PRINT X$
1180 PRINT "EINFRIERTAG.....
:";D$(X)
1190 PRINT "BEMERKUNG:"
1200 PRINT BE$(X)
1210 PRINT :"WEITERSUCHEN=LE
ERTASTE" :: PRINT "ENDE=E"
1220 CALL KEY(0,K,S):: IF S=
0 THEN 1220
1230 IF K=32 THEN 1250
1240 IF K=69 OR K=101 THEN 4
30 ELSE 1220
1250 NEXT X :: PRINT :"NICHT
S MEHR GEFUNDEN." :: FOR A=1
 TO 1000 :: NEXT A :: RETURN
1260 CALL CLEAR :: INPUT "WE
LCHE NUMMER ? ":N
1270 FOR X=1 TO NF
1280 IF NR(X)<>N THEN 1380
1290 CALL CLEAR :: PRINT "VO
R DER AENDERUNG:"
1300 PRINT "KATEGORIE=";KA$(
X):: PRINT "NUMMER= ";NR(X)
1310 PRINT "ART...=..";A$(X)
:: PRINT "STUECK=.";ST(X)
1320 PRINT "GRAMM.=.";GR(X)
1330 GOSUB 2770
1340 PRINT "HALTBARKEITSDAT.
= ";X$
1350 PRINT "EINFRIERDATUM= "
;D$(X)
1360 PRINT "BEMERKUNG: ";BE$
(X):
1370 GOTO 1390
1380 CALL CLEAR :: NEXT X ::
 PRINT "NICHTS MEHR GEFUNDEN
." :: FOR A=1 TO 1000 :: NEX
T A :: RETURN
1390 PRINT :"WAS SOLL GEAEND
ERT WERDEN ? "
1400 PRINT "1=KATEGORIE"
1410 PRINT "2=ART"
1420 PRINT "3=STUECK"
1430 PRINT "4=GRAMM"
1440 PRINT "5=HALTBARKEITSDA
TUM"
1450 PRINT "6=EINFRIERDATUM"
1460 PRINT "7=BEMERKUNG"
1470 PRINT "8=ENDE"
1480 PRINT :"SIE WUENSCHEN ?
 "
1490 CALL KEY(0,CH,S):: IF S
=0 THEN 1490
1500 IF CH<49 OR CH>56 THEN
1480
1510 ON CH-48 GOTO 1520,1530
,1540,1550,1560,1600,1610,43
0
1520 INPUT "DIE NEUE KATEGOR
IE= ":KA$(X):: GOTO 1620
1530 INPUT "DIE NEUE ART= ":
A$(X):: GOTO 1620
1540 INPUT "DIE NEUE STUECKZ
AHL= ":ST(X):: GOTO 1620
```

```
1550 INPUT "DIE NEUE GRAMMZA
HL= ":GR(X):: GOTO 1620
1560 PRINT "NEUES HALTBARKEI
TSDATUM="
1570 INPUT "TAG...":T(X)
1580 INPUT "MONAT.":M(X)
1590 INPUT "JAHR..":J(X):: G
OTO 1620
1600 INPUT "NEUES EINFRIERDA
TUM= ":D$(X):: GOTO 1620
1610 PRINT "NEUE BEMERKUNG="
 :: INPUT BE$(X)
1620 PRINT :"NOCH ETWAS AEND
ERN (J/N)?"
1630 CALL KEY(0,K,S):: IF S=
0 THEN 1630
1640 IF K=74 OR K=106 THEN 1
290 ELSE IF (K<>78 OR K<>110
)=0 THEN 1620
1650 RETURN
1660 CALL CLEAR :: NF=NF+1
1670 PRINT "KATEGORIE:"
1680 PRINT "1=FLEISCH" :: PR
INT "2=GEMUESE" :: PRINT "3=
FISCH" :: PRINT "4=BACKWAREN
" :: PRINT "5=SONSTIGES" ::
PRINT :"SIE WUENSCHEN ?"
1690 CALL KEY(0,AA,S):: IF S
=0 THEN 1690
1700 IF AA<49 OR AA>53 THEN
1690
1710 AA=AA-48 :: IF AA=1 THE
N KA$(NF)="FLEISCH"
1720 IF AA=2 THEN KA$(NF)="G
EMUESE"
1730 IF AA=3 THEN KA$(NF)="F
ISCH"
1740 IF AA=4 THEN KA$(NF)="B
ACKWAREN"
1750 IF AA=5 THEN KA$(NF)="S
ONSTIGES"
1760 NR(NF)=NF :: CALL CLEAR
 :: PRINT "KATEGORIE ";KA$(N
F):: PRINT "NR.=";NR(NF)
1770 INPUT "ART............=
 ":A$(NF)
1780 INPUT "STUECK.........=
 ":ST(NF)
1790 INPUT "GRAMM..........=
 ":GR(NF)
1800 PRINT "HALTBAR BIS ZUM
?"
1810 INPUT "TAG...=":T(NF)
1820 INPUT "MONAT.=":M(NF)
1830 INPUT "JAHR..=":J(NF)
1840 PRINT "EINGEFROREN AM (
TT.MM.JJ)"
1850 INPUT D$(NF)
1860 PRINT "BEMERKUNG ?"
1870 INPUT BE$(NF):: RETURN
1880 CALL CLEAR
1890 PRINT "WELCHER EINTRAG
SOLL" :: PRINT "GELOESCHT WE
RDEN ?"
1900 INPUT "NR. =":N
1910 FOR X=1 TO NF
1920 IF NR(X)<>N THEN 2090
1930 CALL CLEAR :: PRINT "SO
LL DER EINTRAG GELOESCHT" ::
 PRINT "WERDEN ?"
1940 PRINT "KATEGORIE=";KA$(
X):: PRINT "NR.....=";NR(X)
1950 PRINT "ART....= ";A$(X)
:: PRINT "STK....=";ST(X)
1960 PRINT "GRAMM..=";GR(X)
1970 PRINT "EINFRIERDATUM=";
D$(X)
1980 GOSUB 2770
1990 PRINT "HALTBAR BIS..: "
;X$
2000 PRINT "BEMERKUNG="
2010 PRINT BE$(X)
2020 PRINT :: PRINT "EINTRAG
 LOESCHEN (J/N)?"
2030 CALL KEY(0,K,S):: IF S=
0 THEN 2030 :: CALL CLEAR
2040 IF K=74 OR K=106 THEN 2
110 ELSE IF K<>78 OR K<>110
THEN 2030
2050 PRINT "WEITERSUCHEN J O
DER N?"
2060 CALL KEY(0,K,S):: IF S=
0 THEN 2060
2070 IF K=74 OR K=106 THEN 2
090
2080 IF K=78 OR K=110 THEN 4
30 ELSE 2060
2090 NEXT X
2100 PRINT "NICHTS MEHR GEFU
NDEN." :: FOR A=1 TO 1000 ::
 NEXT A :: RETURN
2110 GOSUB 2580 :: RETURN
2120 INPUT "WAS SUCHEN SIE (
ARTIKEL)?...":N$
2130 FOR X=1 TO NF
2140 IF A$(X)=N$ THEN 2170
2150 NEXT X
2160 PRINT :"NICHTS MEHR GEF
UNDEN." :: FOR A=1 TO 1500 :
: NEXT A :: RETURN
2170 CALL CLEAR :: PRINT "KA
TEGORIE=";KA$(X):: PRINT "NR
.....=";NR(X)
2180 PRINT "ART....= ";A$(X)
:: PRINT "STK....=";ST(X)
2190 PRINT "GRAMM..=";GR(X):
: GOSUB 2770
2200 PRINT "HALTBAR BIS..: "
;X$
2210 PRINT "EINFRIERDATUM= "
;D$(X)
2220 PRINT "BEMERKUNG:" :: P
```

```
RINT BE$(X)
2230 PRINT :"WEITER J ODER N
?"
2240 CALL KEY(0,K,S):: IF S=
0 THEN 2240
2250 IF K=74 OR K=106 THEN 2
150
2260 IF K=78 OR K=110 THEN 4
30 ELSE 2240
2270 CALL CLEAR :: PRINT "WA
S MUSS WANN HERAUSGENOMMEN"
:: PRINT "WERDEN ?"
2280 PRINT :"BITTE DATUM EIN
GEBEN.":
2290 INPUT "TAG..=":T1 :: IN
PUT "MONAT=":M1 :: INPUT "JA
HR.=":J1 :: CALL CLEAR
2300 FOR X=1 TO NF
2310 IF T(X)=T1 AND M(X)=M1
AND J(X)=J1 THEN 2450
2320 IF J(X)<J1 THEN 2460
2330 IF T(X)<T1 AND M(X)=M1
AND J(X)=J1 THEN 2460
2340 IF T(X)<T1 AND M(X)=M1
AND J(X)<J1 THEN 2460
2350 IF T(X)<T1 AND M(X)<M1
AND J(X)=J1 THEN 2460
2360 IF T(X)>T1 AND M(X)<M1
AND J(X)=J1 THEN 2460
2370 IF T(X)>T1 AND M(X)=M1
AND J(X)<J1 THEN 2460
2380 IF T(X)<T1 AND M(X)<M1
AND J(X)<J1 THEN 2460
2390 IF T(X)=T1 AND M(X)=M1
AND J(X)<J1 THEN 2460
2400 IF T(X)=T1 AND M(X)<M1
AND J(X)<J1 THEN 2460
2410 IF T(X)=T1 AND M(X)>M1
AND J(X)<J1 THEN 2460
2420 IF T(X)=T1 AND M(X)<M1
AND J(X)=J1 THEN 2460
2430 IF T(X)>T1 AND M(X)>M1
AND J(X)<J1 THEN 2460
2440 GOTO 2570
2450 PRINT "DIESES LEBENSMIT
TEL MUSS" :: PRINT "SOFORT V
ERBRAUCHT WERDEN.": : : :: C
ALL SOUND(800,440,0):: GOSUB
 2770 :: GOTO 2480
2460 PRINT "DIESES  LEBENSMI
TTEL IST ZU" :: PRINT "DEM A
NGEGEBENEN DATUM" :: PRINT "
VERFALLEN.": : : :: CALL SOU
ND(800,440,0)
2470 GOSUB 2770 :: GOTO 2480
2480 PRINT "KATEGORIE:";KA$(
X):: PRINT "NR....=";NR(X)
2490 PRINT "ART...= ";A$(X):
: PRINT "STCK..=";ST(X):: PR
INT "GRAMM.=";GR(X)
2500 GOSUB 2770 :: PRINT "HA
LTBAR BIS:";X$
2510 PRINT "EINFRIERTAG:";D$
(X)
2520 PRINT "BEMERKUNG" :: PR
INT BE$(X): :
2530 PRINT :"WEITERSUCHEN J
ODER N?"
2540 CALL KEY(0,K,S):: IF S=
0 THEN 2540 :: CALL CLEAR
2550 IF K=74 OR K=106 THEN 2
570
2560 IF K=78 OR K=110 THEN 4
30 ELSE 2540
2570 NEXT X :: PRINT :"NICHT
S MEHR GEFUNDEN." :: FOR A=1
 TO 1000 :: NEXT A :: RETURN
2580 FOR X=N TO NF-1
2590 KA$(X)=KA$(X+1):: A$(X)
=A$(X+1):: ST(X)=ST(X+1):: G
R(X)=GR(X+1):: T(X)=T(X+1)::
 M(X)=M(X+1):: J(X)=J(X+1)::
 D$(X)=D$(X+1)
2600 BE$(X)=BE$(X+1):: NEXT
X :: NF=NF-1 :: RETURN
2610 CALL CLEAR
2620 PRINT : :"N E U E I N G
 A B E": : :"SIE KOENNEN NEU
E EINGABEN,":"NACH KATEGORIE
N GETRENNT,IN":"IHRE GEFRIER
TRUHE EINGEBEN.": : : : : :
2630 PRINT : : :"BITTE EINE
TASTE DRUECKEN."
2640 CALL KEY(0,K,S):: IF S=
0 THEN 2640 :: CALL CLEAR
2650 PRINT "A E N D E R N":
: : :"SIE KOENNEN JEDE EINGA
BE":"INNERHALB EINER LAUFEND
EN":"NUMMER AENDERN.NACH DER
 AEN-":"DERUNG WIRD NACH ";
2660 PRINT "WEITEREN ":"AEND
ERUNGEN GEFRAGT.J ODER N": :
 : :"E I N Z E L L O E S C H
 U N-": : :"SIE KOENNEN JEDE
 BELIEBIGE"
2670 PRINT "EINTRAGUNG LOESC
HEN.": :"BITTE EINE TASTE DR
UECKEN."
2680 CALL KEY(0,K,S):: IF S=
0 THEN 2680 :: CALL CLEAR
2690 PRINT "K A T E G O R I
E S U C H E": : : :"HIERBEI
KOENNEN SIE IHRE":"TRUHE NAC
H BESTIMMTEN KATE-":"GORIEN
DURCHSUCHEN.(1-5)": : : :
2700 PRINT "E I N Z E L S U
C H E": : :"BEI DER EINZELSU
CHE KOENNEN":"SIE NACH BESTI
MMTEN ARTEN":"SUCHEN.(KUCHEN
 ODER PIZZA)": : :
2710 PRINT "BITTE EINE TASTE
 DRUECKEN."
```

```
2720 CALL KEY(0,K,S):: IF S=
0 THEN 2720 :: CALL CLEAR
2730 PRINT "G E S A M T I N
H A L T": : :"HIERBEI WIRD D
ER GESAMTE IN-":"HALT IHRER
TRUHE AUF DEM":"BILDSCHIRM O
DER DRUCKER":"AUSGEGEBEN.":
2740 PRINT : : : :"S T I C H
 T A G S U C H E": : :"NACH
EINGABE EINES DATUMS":"SAGT
IHNEN DER RECHNER,":"WELCHE
LEBENSMITTEL VER-"
2750 PRINT "BRAUCHT WERDEN M
UESSEN.": :"BITTE EINE TASTE
 DRUECKEN."
2760 CALL KEY(0,K,S):: IF S=
0 THEN 2760 :: CALL CLEAR ::
 GOTO 430
2770 X$=STR$(T(X))&"."&STR$(
M(X))&"."&STR$(J(X)):: RETUR
N
2780 CALL CLEAR :: PRINT "DA
TEI WIRD SORTIERT."
2790 FOR I=1 TO NF-1 :: FOR
L=1 TO NF-I
2800 IF KA$(L)<KA$(L+1)THEN
2850
2810 H$=KA$(L):: KA$(L)=KA$(
L+1):: KA$(L+1)=H$ :: GOSUB
2860
2820 IF KA$(L)<>KA$(L+1)THEN
 2850
2830 IF A$(L)<A$(L+1)THEN 28
50
2840 GOSUB 2860
2850 NEXT L :: NEXT I :: RET
URN
2860 H$=A$(L):: A$(L)=A$(L+1
):: A$(L+1)=H$ :: H=ST(L)::
ST(L)=ST(L+1):: ST(L+1)=H ::
 H=GR(L):: GR(L)=GR(L+1):: G
R(L+1)=H
2870 H=T(L):: T(L)=T(L+1)::
T(L+1)=H :: H=M(L):: M(L)=M(
L+1):: M(L+1)=H :: H=J(L)::
J(L)=J(L+1):: J(L+1)=H
2880 H$=D$(L):: D$(L)=D$(L+1
):: D$(L+1)=H$ :: H$=BE$(L):
: BE$(L)=BE$(L+1):: BE$(L+1)
=H$ :: RETURN
2890 END
```

BRUCH-RECHNEN

Ab dem 6. Schuljahr geht es los. Die vielerseits ungeliebte Bruchrechnung wird in den Lehrplan mit aufgenommen. Den Eltern graust es schon bei dem Gedanken, die Hausaufgaben der Pennäler kontrollieren zu müssen und die Schüler selbst werden sich wünschen, daß dieser Unterrichtsstoff bald abgehandelt ist und wieder vom Lehrplan verschwindet.

Doch Bruchrechnen ist ein wichtiges Thema und jeder muß leider damit leben.

Um den TI 99/4A-Besitzern den Umgang mit der Bruchrechnung etwas zu erleichtern, hat der Autor dieses Programm geschrieben (sogar in normalem TI-Basic, damit auch wirklich jeder von den Bürden der Bruchrechnung entlastet wird).

Nach dem Abtippen dieses Listings wird sicherlich ein allgemeines Aufatmen durch die Menge gehen. Eltern können nun schneller und sicherer die Aufgaben ihrer Sprößlinge kontrollieren, auch wenn für sie das Thema „Bruchrechnen" schon seit einigen Jahren erledigt ist und die ganzen Rechenregeln längst wieder vergessen sind.

Für die Schüler selbst bietet dieses Programm sogar noch eine weitere Möglichkeit an. Sollten sie im Unterricht nicht ganz mitgekommen sein, können sie sich von ihrem TI Übungsaufgaben in zwei verschiedenen Schwierigkeitsgraden stellen lassen.

Der TI 99 kann zwar im Befehlsmodus selbst eine ganze Menge verschiedener Rechnungen ausführen, doch für die Bruchrechnung ist der Command-Mode nicht gerade geeignet, da er keine Ergebnisse als Bruchzahlen wieder gibt.

Im Gegensatz dazu gibt dieses Programm nicht nur die Ergebnisse als Bruchzahl aus, sondern liefert auch noch gleich die Zwischenergebnisse mit. Mancher Lehrer soll ja bekanntlich gerade auf die Zwischenergebnisse großen Wert legen, um zu sehen, ob die Gedankengänge richtig waren oder auch nur um zu kontrollieren, ob das Programm nicht abgeschrieben wurde.

```
BRUCHRECHNEN

FUER ERGEBNIS=LEERTASTE
DRUECKEN.

1=ADDIEREN
2=SUBTRAHIEREN
3=MULTIPLIZIEREN
4=DIVIDIEREN

WELCHE FUNKTION?
```

Das Menue

Nun zur Programmsteuerung:

Nach dem Starten des Programmes erscheint auf

Bitte lesen Sie weiter auf Seite 49

```
100 REM*********************
110 REM*                   *
120 REM*   Bruchrechnen    *
130 REM*   Copyright by    *
140 REM*     B.Knedel      *
150 REM*                   *
210 REM*Benoetigte Geraete*
220 REM* TI99/4A  Konsole *
230 REM*                   *
240 REM*                   *
250 REM* Speicherbelegung *
260 REM*    4837 Bytes     *
270 REM*********************
280 REM
290 CALL SCREEN(16)
300 RANDOMIZE
310 CALL CLEAR
320 H$="ZWISCHENERGEBNIS:"
330 CALL CHAR(128,"00000055A
A55")
340 PRINT ".        BRUCHRECHN
EN": : : : : : :"SCHWIERIGKE
ITSGRAD 1 ODER 2?": : : : :
:
350 PRINT "GEMISCHTE ZAHLEN
VOR DER": :"EINGABE BITTE IN
 BRUECHE UM-": :"WANDELN.":
:
360 CALL KEY(0,P,S)
370 IF S=0 THEN 360
380 IF (P<49)+(P>50)=-1 THEN
 360
390 IF P=49 THEN 420
400 T=19
410 GOTO 430
420 T=9
430 CALL CLEAR
440 PRINT ".......BRUCHRECHN
EN": : : : :"FUER ERGEBNIS=L
EERTASTE": :"DRUECKEN.": : :
 :
450 PRINT "SOLL DER COMPUTER
 AUFGABEN": :"STELLEN?": : :
 : : : :"J ODER N ?"
460 CALL KEY(0,J,S)
470 IF (J<>74)+(J<>78)+(J<>1
06)+(J<>110)=-4 THEN 460
480 IF (J=74)+(J=106)THEN 57
0
490 CALL CLEAR
500 PRINT "........BRUCHRECH
NEN": : : :"FUER ERGEBNIS=LE
ERTASTE": :"DRUECKEN.": : :
: :
510 PRINT "1=ADDIEREN": :"2=
SUBTRAHIEREN": :"3=MULTIPLIZ
IEREN": :"4=DIVIDIEREN": : :
520 PRINT "WELCHE FUNKTION?"
530 CALL KEY(0,R,S)
540 IF S=0 THEN 530
550 IF (R<49)+(R>52)<>0 THEN
 530
560 GOTO 670
570 CALL SOUND(500,440,1)
580 Z=INT(RND*T)+1
590 N=INT(RND*T)+1
600 Z1=INT(RND*T)+1
610 N1=INT(RND*T)+1
620 IF Z/N<=Z1/N1 THEN 580
630 IF (N=1)+(N1=1)<>0 THEN
580
640 IF (Z=N)+(Z1=N1)<>0 THEN
 580
650 R=INT(RND*4)+1
660 GOTO 690
670 CALL CLEAR
680 R=R-48
690 IF R=1 THEN 740
700 IF R=2 THEN 760
710 IF R=3 THEN 780
720 C=58
730 GOTO 790
740 C=43
750 GOTO 790
760 C=45
770 GOTO 790
780 C=42
790 IF (J=74)+(J=106)THEN 88
0
800 CALL HCHAR(22,4,C,4)
810 INPUT "ZAEHLER 1 ? ":Z
820 PRINT
830 INPUT "NENNER..1 ? ":N
840 PRINT : : :
850 INPUT "ZAEHLER 2 ? ":Z1
860 PRINT
870 INPUT "NENNER..2 ? ":N1
880 GOSUB 2030
890 GOSUB 2090
900 GOSUB 2200
910 GOSUB 2250
920 IF (R=2)+(SGN(Z/N-Z1/N1)
=-1)=-2 THEN 940
930 GOTO 960
940 CALL SOUND(1000,-6,1)
950 GOTO 430
960 Z=VAL(A$)
970 N=VAL(B$)
980 Z1=VAL(A1$)
990 N1=VAL(B1$)
1000 X=N
1010 Y=N1
1020 IF X=Y THEN 1080
1030 IF X<Y THEN 1060
1040 Y=Y+N1
1050 GOTO 1020
1060 X=X+N
1070 GOTO 1020
1080 KGV=X
1090 ON R GOTO 1600,1690,1780
```

```
1100 CALL KEY(0,P,S)
1110 CALL SOUND(-30,440,10)
1120 IF S=0 THEN 1100
1130 GOSUB 2490
1140 IF Z=GGT THEN 1420
1150 IF Z<GGT THEN 1160 ELSE
 1190
1160 A$=STR$(Z)
1170 B$=STR$(GGT)
1180 GOTO 1300
1190 X=INT(Z/GGT)
1200 Z=Z-GGT*X
1210 A$=STR$(X)
1220 IF (Z=0)+(N=1)<>0 THEN
1260
1230 B$=STR$(Z)
1240 C$=STR$(GGT)
1250 GOTO 1450
1260 FOR I=1 TO LEN(A$)
1270 CALL HCHAR(13,L+L1+I+6,
ASC(SEG$(A$,I,1)))
1280 NEXT I
1290 GOTO 2370
1300 FOR I=1 TO LEN(A$)
1310 CALL HCHAR(12,6+L+I+L1,
ASC(SEG$(A$,I,1)))
1320 NEXT I
1330 FOR I=1 TO LEN(B$)
1340 CALL HCHAR(14,6+L+L1+I,
ASC(SEG$(B$,I,1)))
1350 NEXT I
1360 IF LEN(A$)>LEN(B$)THEN
1390
1370 L2=LEN(B$)
1380 GOTO 1400
1390 L2=LEN(A$)
1400 CALL HCHAR(13,7+L+L1,12
8,L2)
1410 GOTO 2370
1420 CALL HCHAR(13,6+L1+L2,1
28)
1430 CALL HCHAR(13,7+L+L1,49
)
1440 GOTO 2370
1450 FOR I=1 TO LEN(A$)
1460 CALL HCHAR(13,6+L+L1+I,
ASC(SEG$(A$,I,1)))
1470 NEXT I
1480 FOR I=1 TO LEN(B$)
1490 CALL HCHAR(12,6+L+L1+LE
N(A$)+I,ASC(SEG$(B$,I,1)))
1500 NEXT I
1510 FOR I=1 TO LEN(C$)
1520 CALL HCHAR(14,6+L+L1+LE
N(A$)+I,ASC(SEG$(C$,I,1)))
1530 NEXT I
1540 IF LEN(C$)>LEN(B$)THEN
1570
1550 L2=LEN(B$)
1560 GOTO 1580
1570 L2=LEN(C$)
1580 CALL HCHAR(13,7+L+LEN(A
$)+L1,128,L2)
1590 GOTO 2370
1600 QQ=Z1
1610 Q=Z
1620 Z=KGV/N*Z
1630 Z1=KGV/N1*Z1
1640 Z=Z+Z1
1650 Z5=Q*N1+N*QQ
1660 N5=N*N1
1670 GOSUB 1900
1680 GOTO 1100
1690 PP=Z
1700 PP1=Z1
1710 Z=KGV/N*Z
1720 Z1=KGV/N1*Z1
1730 Z=Z-Z1
1740 N5=N*N1
1750 Z5=N5/N*PP-N5/N1*PP1
1760 GOSUB 1900
1770 GOTO 1100
1780 Z=Z*Z1
1790 X=N*N1
1800 Z5=Z
1810 N5=X
1820 GOSUB 1900
1830 GOTO 1100
1840 Z=Z*N1
1850 X=Z1*N
1860 Z5=Z
1870 N5=X
1880 GOSUB 1900
1890 GOTO 1100
1900 A=Z
1910 B=X
1920 IF A=B THEN 1980
1930 IF A>B THEN 1960
1940 B=B-A
1950 GOTO 1920
1960 A=A-B
1970 GOTO 1920
1980 Z=Z/A
1990 Z6=Z
2000 GGT=X/A
2010 N6=GGT
2020 RETURN
2030 A$=STR$(Z)
2040 CALL CLEAR
2050 FOR I=1 TO LEN(A$)
2060 CALL HCHAR(12,4+I,ASC(S
EG$(A$,I,1)))
2070 NEXT I
2080 RETURN
2090 B$=STR$(N)
2100 FOR I=1 TO LEN(B$)
2110 CALL HCHAR(14,4+I,ASC(S
EG$(B$,I,1)))
2120 NEXT I
2130 IF LEN(A$)>LEN(B$)THEN
```

```
216Ø
214Ø L=LEN(B$)
215Ø GOTO 217Ø
216Ø L=LEN(A$)
217Ø CALL HCHAR(13,5,128,L)
218Ø CALL HCHAR(13,5+L,C)
219Ø RETURN
22ØØ A1$=STR$(Z1)
221Ø FOR I=1 TO LEN(A1$)
222Ø CALL HCHAR(12,5+L+I,ASC
(SEG$(A1$,I,1)))
223Ø NEXT I
224Ø RETURN
225Ø B1$=STR$(N1)
226Ø FOR I=1 TO LEN(B1$)
227Ø CALL HCHAR(14,5+L+I,ASC
(SEG$(B1$,I,1)))
228Ø NEXT I
229Ø IF LEN(A1$)>LEN(B1$)THE
N 232Ø
23ØØ L1=LEN(B1$)
231Ø GOTO 233Ø
232Ø L1=LEN(A1$)
233Ø CALL HCHAR(13,6+L,128,L
1)
234Ø CALL HCHAR(13,6+L+L1,61
)
235Ø IF R=4 THEN 184Ø
236Ø RETURN
237Ø PRINT "NOCH EINMAL J OD
ER N?"
238Ø CALL KEY(Ø,K,S)
239Ø IF (K=78)+(K=11Ø)=-1 TH
EN 246Ø
24ØØ IF (K<>74)+(K<>1Ø6)<>-1
 THEN 238Ø
241Ø X=Ø
242Ø Y=Ø
243Ø GGT=Ø
244Ø KGV=Ø
245Ø GOTO 43Ø
246Ø CALL CLEAR
247Ø PRINT "AUF WIEDERSEHEN.
": : : : :"SCHOENEN GRUSS VO
N": : : : : : :"........ADAM
 RIESE!": : : : : : : :
248Ø END
249Ø FOR I=1 TO LEN(H$)
25ØØ CALL HCHAR(2,4+I,ASC(SE
G$(H$,I,1)))
251Ø NEXT I
252Ø Z5$=STR$(Z5)
253Ø N5$=STR$(N5)
254Ø Z6$=STR$(Z6)
255Ø N6$=STR$(N6)
256Ø IF LEN(Z5$)>LEN(N5$)THE
N 259Ø
257Ø LA=LEN(N5$)
258Ø GOTO 26ØØ
259Ø LA=LEN(Z5$)
26ØØ FOR I=1 TO LEN(Z5$)
261Ø CALL HCHAR(4,4+I,ASC(SE
G$(Z5$,I,1)))
262Ø NEXT I
263Ø CALL HCHAR(5,5,128,LA)
264Ø FOR I=1 TO LEN(N5$)
265Ø CALL HCHAR(6,4+I,ASC(SE
G$(N5$,I,1)))
266Ø NEXT I
267Ø IF LEN(Z6$)>LEN(N6$)THE
N 27ØØ
268Ø LA1=LEN(N6$)
269Ø GOTO 271Ø
27ØØ LA1=LEN(Z6$)
271Ø FOR I=1 TO LEN(Z6$)
272Ø CALL HCHAR(4,5+LA+I,ASC
(SEG$(Z6$,I,1)))
273Ø NEXT I
274Ø CALL HCHAR(5,6+LA,128,L
A1)
275Ø FOR I=1 TO LEN(N6$)
276Ø CALL HCHAR(6,5+LA+I,ASC
(SEG$(N6$,I,1)))
277Ø NEXT I
278Ø RETURN
```

Fortsetzung von Seite 46

BRUCH-RECHNEN

dem Bildschirm ein Menü. Zum einen kann man jetzt eine der vier Grundrechenarten anwählen, mit der die Rechenaufgabe gelöst werden soll und zum anderen möchte der Computer gerne wissen, ob er mit Hilfe seines Zufallgenerators Aufgaben stellen oder vorgegebene Aufgaben lösen soll.
In beiden Fällen tut er die Beendigung seiner Aufgabe mit einem Piepston kund, ohne das Zwischenergebnis und die Endsumme auf dem Bildschirm auszugeben.
Dies soll Euch die Zeit verschaffen, die Ihr benötigt, um die Aufgabe selbst zu lösen. Habt Ihr nun auch ein Ergebnis zu Papier gebracht (oder auch nicht), genügt es vollkommen, wenn Ihr jetzt einmal die Leertaste betätigt. Der TI sieht sich nun veranlaßt, seine Zwischenergebnisse und die Endsumme auszugeben.
Der Übungsteil dieses Programmes beinhaltet zwei verschiedene Schwierigkeitsstufen. Die erste für blutige Laien auf dem Gebiet der Brechrechnung und die zweite für Schüler (oder Eltern) die schon etwas tiefer in die Materie eingestiegen sind.

```
ZWISCHENERGEBNIS:
105  21
---  --
200  40

15   7   21
-- * - = --
25   8   40
```

Zwei Einschränkungen müssen bei diesem Programm leider beachtet werden:
1.) Es dürfen keine gemischten Brüche eingegeben werden. Gemischte Brüche müssen vor der Eingabe erst in echte Brüche umgewandelt werden. Dies dürfte aber wohl für jeden eine Kleinigkeit sein.
2.) Kann das Programm keine Kettenbrüche lösen. Wenn man aber in Zweierschritten vorgeht, kommt man auf alle Fälle trotzdem schneller zu einem richtigen Ergebnis, als dies mit Füller und Papier zu erreichen wäre.

Die Eingabe der Brüche erfolgt in vier Schritten im Dialog mit dem TI:
1.) Der Zähler des ersten Bruches.
2.) Der Nenner des ersten Bruches.
3.) Der Zähler des zweiten Bruches.
4.) Der Nenner des zweiten Bruches.

Die Rechenart wird ja bereits im Hauptmenü ausgewählt und braucht nun nicht weiter beachtet werden.

Mehr ist bei diesem Programm eigentlich nicht zu beachten. Die Bedienung ist kinderleicht und durch das Menü und den programmierten Dialog können eigentlich keine Fehlbedienungen und damit vielleicht falsche Ergebnisse auftreten.

Auch Lehrern (es gibt ja einige, die einen TI besitzen) könnte dieses Programm eine Hilfe sein, wenn sie zu Hause die Übungsaufgaben für den nächsten Tag zusammenstellen und sich etwas Zeit einsparen wollen. Dadurch, daß das Programm auch die Zwischenergebnisse mit dem kleinsten gemeinsamen Nenner liefert, ist auch die Lernkontrolle gewährleistet.

Schüler, die mit der Bruchrechnung auf Kriegsfuß stehen, sollten sich unbedingt mit dem Übungsteil dieses Programmes befassen. Er macht mehr Spaß, als die sture Rechnerei mit Füller und Papier und liefert auch auf Anforderung gleich die richtigen Ergebnisse, damit sie sehen können, ob Sie alles gefressen haben.

Wer die nächste Mathe-Arbeit nun dennoch verhaut, ist selbst dran Schuld.

Hartmut König

Und es geht doch: Maschinenprogramme von Kassette laden mit dem Extended Basic-Modul

Assembler-Programme — in assemblierter Form auch Maschinenprogramme genannt — werden in der Regel von einer Diskette in den Computer geladen. Und wer keine Diskettenstation hat, guckt in die Röhre — meint man. Stimmt aber nicht, denn der im Extended-Basic-Modul enthaltene Loader kann Maschinenprogramme auch von der Kassette laden! Einen Haken hat die Sache aber: Es gibt weder Angebote von Maschinenprogrammen auf Kassette noch kann das Editor/Assembler-Modul seinen Output auf Kassette schreiben. Also wurde von einer nutzlosen Fähigkeit des Extended-Basic-Moduls gesprochen? Nein!

In diesem Artikel wird beschrieben, wie man — mit Hilfe eines Freundes, der glücklicher Besitzer einer Diskettenstation ist — Kassetten mit Maschinenprogrammen beschreiben kann, die dann das Extended-Basic-Modul akzeptiert.

Alle Bedingungen, die der XB-Loader an Diskettenprogramme stellt, müssen natürlich auch von den Kassettenprogrammen erfüllt werden: Der 'tagged object code' darf nicht 'compressed' sein (eine besondere Option des E/A-Moduls, die Platz auf dem Datenträger spart) und er darf keine externen Referenzen (REFs) enthalten. Wenn aber ein fertiges Programm auf Diskette für das Extended-Basic-Modul bestimmt ist, dann werden diese Forderungen sicherlich erfüllt sein.

Es genügt nun aber nicht, für die Konversion einfach die Datensätze von der Diskette zu lesen und 1:1 auf die Kassette zu übertragen. Dafür sind sie zu lang. Die Datensätze auf der Diskette haben ein festes Format von 80 Bytes Länge, die auf der Kassette dürfen nur 64 lang sein. Die Datensätze müssen also umgeordnet werden. Das Zerschneiden darf jeweils nur vor einer 'tag' genannten Marke erfolgen. Die Satzabschlüsse mit 'checksum tag', 'checksum' und 'end sentinel' müssen entfernt und nach Umgruppierung der Datensätze wieder richtig angefügt werden. Für „Eingeweihte", die das Konversionsprogramm ergründen möchten: Es wird dann das checksum tag 8 verwendet, das beim Laden die Kontrolle der checksum unterbindet. Man braucht also nicht eine neue Prüfsumme zu berechnen. Und der Anschein, den der Original-Assembler-Output erweckt, trügt, der nämlich, daß jeder Datensatz mit einer Ladeadresse beginnen muß. Das ist nicht nötig, und deshalb ist das Umgruppieren wieder ein Stück einfacher. Die besonderen Datensätze, die beispielsweise die Namen von Programm-Marken mit den zugehörigen Adressen (DEFs) übermitteln oder die das Programmende anzeigen, müssen trotz der Umgruppierung ihr spezielles Format beibehalten.

Das abgedruckte Extended-Basic-Programm erledigt die ganze Konversion. Seine sehr einfache Handhabung wird jetzt beschrieben. Es fragt anfangs nach dem File-Namen des Diskettenprogramms, das auf Kassette übertragen werden soll, z.B. DSK1.SPIELX.

Dann kommen die üblichen Anweisungen zur Bedienung des Kassettenrekorders, und es wird Satz für Satz auf die Kassette geschrieben. Bei der Konversion auftretende Fehler werden am Monitor gemeldet. Das Programm endet — hoffentlich! — mit der Meldung 'Konversion fertig'.

Wer sich dafür interessiert, wie die umgruppierten Datensätze auf dem Band aussehen, kann anschließend mit RUN 600 einen Ausdruck (über PIO, nötigenfalls ändern!) erzeugen.

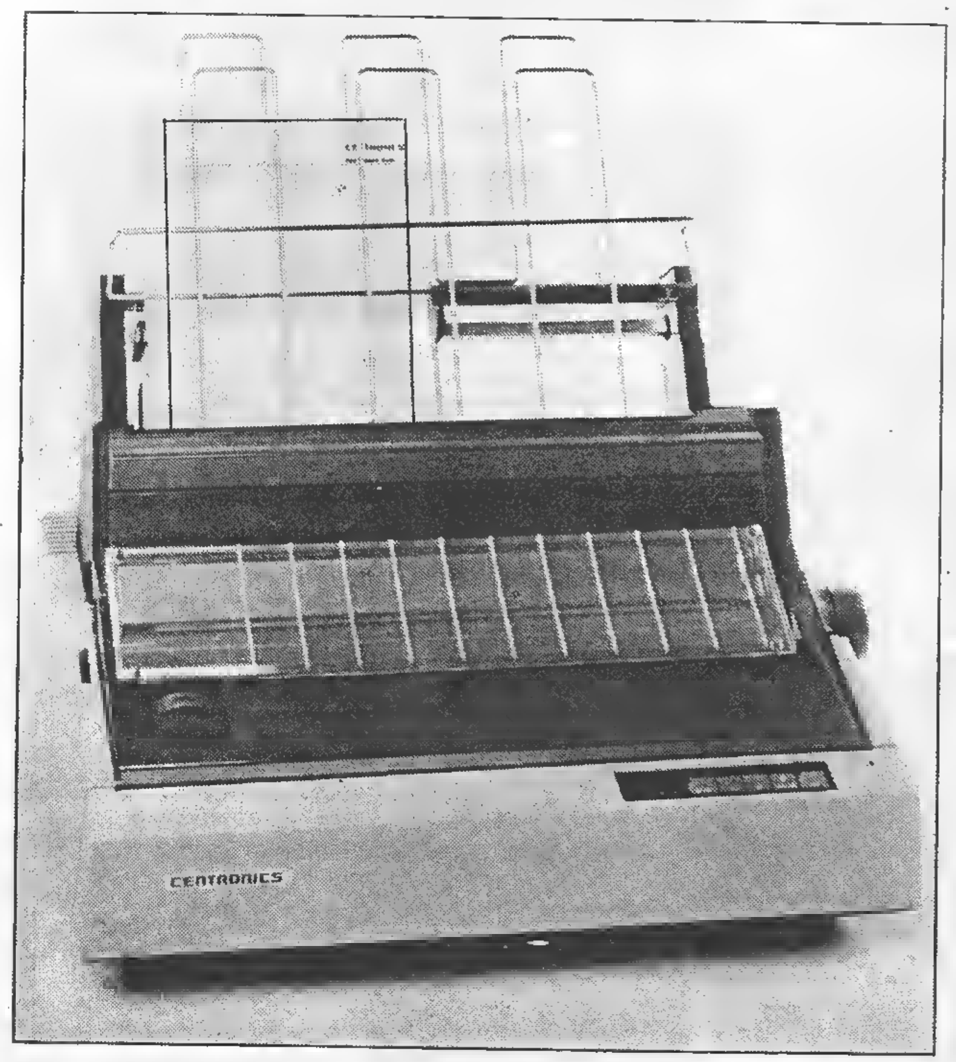
Dann kann geladen werden: Nach CALL INIT und CALL LOAD("CS1") geht's los mit CALL LINK("START"), wobei angenommen ist, daß der Startname ('Entry name') START ist.

Gutes Gelingen! *W. Jüngst*

```
100 !-----------------------
110 ! KONVERSION VON
120 ! OBJECT-CODE FUER
130 ! KASSETTENBENUTZER
140 !
150 ! VORHER:
160 ! E/A-OUTPUT OHNE REFs
170 ! AUF DISKETTE IM
180 ! FORMAT DIS/FIX 80,
190 ! NON-COMPRESSED
200 !
210 ! NACHHER:
220 ! KONVERTIERTER OUTPUT
230 ! AUF KASSETTE IM FORMAT
240 ! DIS/FIX 64, GEEIGNET
250 ! FUER DEN XB-LOADER MIT
260 ! CALL LOAD("CS1")
270 !
280 ! AUTOR: W.JUENGST
320 F$="80000F" :: ER$="* FE
HLER:" :: PRINT :: INPUT "DI
SK-OBJ-FILE-NAME: ":FN$
330 OPEN #1:FN$,FIXED 80,INP
UT :: OPEN #2:"CS1",FIXED 64
,OUTPUT
340 INPUT #1:A$ :: GOSUB 510
350 IF L<=64 THEN R=0 :: R$=
"" :: PRINT #2:SEG$(A$,1,L-6
)&F$ :: GOTO 370
360 R=5 :: R$=SEG$(A$,59,5):
: PRINT #2:SEG$(A$,1,58)&F$
370 FOR Z=2 TO 10000 :: INPU
T #1:A$ :: AS=ASC(A$):: GOSU
B 510
380 IF AS=65 OR AS=57 OR AS=
66 OR AS=67 THEN 450
390 IF R$<>"" THEN PRINT #2:
R$&F$ :: R$="" :: R=0
400 IF AS=49 OR AS=50 THEN P
RINT :"* AUTOSTART IN XB IGN
ORIERT" :: GOTO 490
410 IF AS=58 THEN PRINT #2:S
EG$(A$,1,L):: PRINT :"*.FERT
IG KONV." :: GOTO 500
420 IF AS=51 OR AS=52 OR AS=
53 OR AS=54 THEN 430 ELSE PR
INT :ER$ :: GOTO 500
430 IF L<=64 THEN PRINT #2:S
EG$(A$,1,L-6)&F$ :: GOTO 490
440 PRINT #2:SEG$(A$,1,50)&F
$ :: PRINT #2:SEG$(A$,51,10)
&F$ :: GOTO 490
450 A$=R$&SEG$(A$,1,L-6):: L
=L+R-6 :: R$="" :: R=0
460 IF L<=58 THEN PRINT #2:A
$&F$ :: GOTO 490
470 PRINT #2:SEG$(A$,1,55)&F
$ :: R=L-55 :: R$=SEG$(A$,56
,R):: IF R<51 THEN 490
480 PRINT #2:R$&F$ :: R$=""
:: R=0
490 NEXT Z
500 CLOSE #1 :: CLOSE #2 ::
STOP
510 FOR L=69 TO 1 STEP -1 ::
 IF SEG$(A$,L,1)<>" " THEN R
ETURN
520 NEXT L :: RETURN
600 OPEN #2:"CS1",FIXED 64,I
NPUT :: OPEN #3:"PIO"
610 FOR I=1 TO 10000 :: INPU
T #2:A$ :: PRINT #3:A$
620 IF ASC(A$)=58 THEN CLOSE
 #2 :: CLOSE #3 :: STOP
630 NEXT I
```

Machen Sie Druck beim Drucken

Bekanntlich steuert man die Vielzahl des prinzipiell vorhandenen Druckkomforts bei einem Matrixdrucker mittels Stringzeichenfunktion (CHR$) in Verbindung mit einer PRINT-Anweisung an. Dabei unterliegen die Codes der Charaktere 1 – 29 einer Normung, sind also im Prinzip bei allen Druckern gleich. Bei hochwertigen Druckern reicht aber diese Anzahl von Einzelsteuerzeichen für die Realisierung der Vielzahl der unterschiedlichsten Druckleistungen nicht aus. Deshalb erfolgt ein Großteil der Ansteuerung des Druckers vom Computer aus, mittels sogenannter Escape-Sequenzen.



Escape (symbolische Abkürzung ESC) bedeutet wörtlich übersetzt „entschlüpfen", „entkommen". Da es sich hierbei um einen Hilfscode handelt, dem mindestens ein weiteres Steuerzeichen folgt, spricht man auch von Einleitungssteuerzeichen, oder wenn der gesamte symbolische Code gemeint ist, von Fluchtsequenzen (Escape-Sequenzen oder auch ESC-Sequenzen). Bei der Benutzung von Fluchtsequenzen ist folgendes unbedingt zu beachten:

- ☐ In den Bedienungsanleitungen für Matrixdrucker erfolgt meist die Formulierung der Sequenzen symbolisch. In dieser Weise lassen sie sich aber in keinem Fall in den Computer eingeben.
- ☐ Die ESC-Sequenzen unterliegen keiner Normung. Deshalb müssen gleichartig geschriebene symbolische Codes bei unterschiedlichen Druckern keinesfalls die gleiche Wirkung des Drucks hervorrufen.
- ☐ Die computergerechte Verschlüsselung des Einleitungssteuerzeichen ESC geschieht in jedem Fall mittels Stringzeichenfunktion und zwar genormt mit dem Code 27. Also anstatt ESC gilt automatisch immer CHR$(27).
- ☐ Die Verschlüsselung der dem Einleitungssteuerzeichen folgenden (symbolischen) Code – ein Buchstabe, eine Ziffer oder ein Sonderzeichen – kann auf unterschiedliche Weise realisiert werden.

VERSCHLÜSSELUNG EINER ESC-SEQUENZ IN VERBINDUNG MIT DEM TI 99/4A

Um keine Irrtümer aufkommen zu lassen, sei daran erinnert, daß bei Übermittlung einer Fluchtsequenz an den Drucker ein Druckkanal offen sein muß (Open-Anweisung). Wir gehen im folgenden davon aus, daß die logische Filenummer 1 angesprochen wurde. Die Bedienungsanleitung für den MX-80 (Typ III) – gilt in gleicher Weise auch für den RX-80 und den FX-80 – gibt beispielsweise für die Druckart Fett folgenden symbolischen Code an: "ESC E"-Code oder in der Anleitung für den RX-80 bzw. FX-80 einfach den Namen ESC E:
Für den TI 99/4A gibt es vier Varianten, die hier als Beispiel ausgewählte Escape-Sequenzen in Form einer PRINT-Anweisung an den Drucker zu übermitteln:

- O PRINT #1:CHR$(27); "E" oder
- O PRINT #1:CHR$(27);CHR$(69) oder
- O PRINT #1:CHR$(27)&CHR$(69) oder
- O PRINT #1:CHR$(27)&"E"

Die gewählte Form im zweiten und dritten Beispiel ist manchmal praktikabler, weil der Code innerhalb der zweiten Stringzeichenfunktion auch als numerischer Ausdruck gestattet ist. Also nachfolgende Anweisung würde ebenfalls den Drucker auf Fettdruck umschalten.

```
CODE = 69
PRINT #1:CHR$(27);CHR$
(CODE)
```

An Stelle des Semikolon kann natürlich auch ein Stringverknüpfungszeichen stehen (&).
Wird die ESC-Sequenz als Stringvariable vereinbart und auf diese Weise innerhalb des Programms benutzt, so ist aus Gründen der Syntax nur beispielsweise folgendes möglich:

```
CODE$=CHR$(27)&"E"    oder
CODE$=CHR$ (27)&CHR$(69)
```

Folgt dem zweiten symbolischen Code noch ein weiteres Steuerzeichen, so handelt es sich in jedem Fall um einen Parameter, mit dessen Hilfe eine Spezifizierung der betroffenen ESC-Sequenz erfolgt. Dieser Parameter wird symbolisch meist mit n angegeben. n ist hierbei eine numerische Variable, die dezimal eingebunden in eine Stringzeichenfunktion endgültig zu formulieren ist. Die symbolische Darstellung

"ESC A+(n)D"-Code muß demnach computergerecht wie folgt umgesetzt werden, wenn n den Wert 26 erhalten soll:

PRINT #1:CHR$(27);"A";
CHR$(26) oder als indizierte Stringvariable

CODE$(19)=CHR$(27)&"A"
&CHR$(26)

Bei den drei genannten Druckern bewirkt die genannte PRINT-Anweisung, daß der Zeilenabstand auf 26/72 Inch festgesetzt wird. Der Parameter n darf in unserem Beispiel dabei einen Wert von 1 bis 85 erhalten. n entspricht hier exakt formuliert der Anzahl der Nadeln zwischen den Zeilen. Natürlich gelten für die Formulierung der PRINT-Anweisung ebenfalls die vorher genannten Varianten. Somit ist auch folgendes möglich:

N=13
PRINT #1:CHR$(27);CHR$(65);
CHR$(N)

Der Zeilenabstand wird auf 13/72 Inch festgelegt.
Die Verschlüsselung der ESC-Sequenz, die eine vorher eingestellte Druckwirkung wieder aufheben soll—

```
100 REM DEMONSTRATION MIT TI
-99/4A
105 REM CTRL + ZEICHEN ERZEU
GT ESC-SEQUENZ
110 REM GETESTET AUF EPSON M
X-80 (TYP III)
115 REM -------------------
----------------------------
120 REM UMSCHLUESSELUNGSLIST
E = 1. SCHRITT
125 REM -------------------
----------------------------
130 DIM CODE$(18)
135 CODE$(1)=CHR$(9)
140 REM HT      -HORIZONTALEN
TABULATOR ANLAUFEN     CTRL A
145 CODE$(2)=CHR$(27)&"0"
150 REM ESC 0 -ZEILENABSTAND
 1/8 INCH =DICHT       CTRL B
155 CODE$(3)=CHR$(27)&"1"
160 REM ESC 1 -ZEILENABSTAND
 7/72 INCH=SEHR DICHT CTRL C
165 CODE$(4)=CHR$(27)&"2"
170 REM ESC 2 -ZEILENABSTAND
 1/6 INCH =NORMAL      CTRL D
175 CODE$(5)=CHR$(27)&"E"
180 REM ESC E -FETTDRUCK EIN
SCHALTEN               CTRL E
185 CODE$(6)=CHR$(27)&"F"
190 REM ESC F -FETTDRUCK AUS
SCHALTEN               CTRL F
195 CODE$(7)=CHR$(27)&"G"
200 REM ESC G -DOPPELDRUCK E
INSCHALTEN             CTRL G
205 CODE$(8)=CHR$(27)&"H"
210 REM ESC H -DOPPELDRUCK A
USSCHALTEN             CTRL H
215 CODE$(9)=CHR$(27)&"T"
220 REM ESC T -VERKLEINERTEN
 DRUCK AUSSCHALTEN     CTRL I
225 CODE$(10)=CHR$(27)&"@"
230 REM ESC @ -ZEICHENSATZ I
TALIEN AUS/NORMIERUNG CTRL J
235 CODE$(11)=CHR$(27)&"D"&C
HR$(5)
240 REM ESC D(5) -HT/TABULAT
OR AUF 5.POS. SETZEN  CTRL K
245 CODE$(12)=CHR$(27)&"R"&C
HR$(6)
250 REM ESC R(6) -ZEICHENSAT
Z ITALIEN WAEHLEN      CTRL L
255 CODE$(13)=CHR$(27)&"S"&C
HR$(0)
260 REM ESC S(0) -VERKLEINER
TEN DRUCK HOCHSTELLEN CTRL M
265 CODE$(14)=CHR$(27)&"S"&C
HR$(1)
270 REM ESC S(1) -VERKLEINER
TEN DRUCK TIEFSTELLEN CTRL N
275 CODE$(15)=CHR$(27)&"W"&C
HR$(1)
280 REM ESC W(1) -GEDEHNTE S
CHRIFT EINSCHALTEN     CTRL O
285 CODE$(16)=CHR$(27)&"W"&C
HR$(0)
290 REM ESC W(0) -GEDEHNTE S
CHRIFT AUSSCHALTEN     CTRL P
295 CODE$(17)=CHR$(27)&"-"&C
HR$(1)
300 REM ESC -(1) -UNTERSTREI
CHUNG EINSCHALTEN      CTRL Q
305 CODE$(18)=CHR$(27)&"-"&C
HR$(0)
310 REM ESC -(0) -UNTERSTREI
CHUNG AUSSCHALTEN      CTRL R
315 REM -------------------
----------------------------
320 REM SCHREIBEN EINER TEXT
SEITE = 2. SCHRITT
325 REM -------------------
----------------------------
330 DIM A$(30)
335 FOR Z=1 TO 30
340 CALL CLEAR
345 PRINT " MIT '!' TEXTEING
ABE BEENDEN"
350 PRINT
355 PRINT "LETZTE ZEILE (";Z
-1;".) LAUTET:"
360 PRINT
365 PRINT "   ";A$(Z-1)
```

```
370 PRINT
375 PRINT
38Ø PRINT "MAXIMAL ZWEI ZEIL
EN EINGEBEN"
385 PRINT
39Ø INPUT A$(Z)
395 IF A$(Z)="!" THEN 4Ø5
4ØØ NEXT Z
4Ø5 CALL CLEAR
41Ø PRINT "      TEXT GESCHR
IEBEN"
415 PRINT
42Ø I=Z
425 REM UEBERNAHME DER GESCH
RIEBENEN ZEILENANZAHL
43Ø REM -------------------
----------------------------
435 REM CTRL ZEICHEN GEGEN S
TEUERCODE = 3. SCHRITT
44Ø REM -------------------
----------------------------
445 PRINT "  AUSTAUSCH ERFOL
GT JETZT !"
45Ø PRINT
455 PRINT "       BITTE WART
EN"
46Ø FOR Z=1 TO I-1
465 REM ZEILE FUER ZEILE
47Ø L=LEN(A$(Z))
475 IF L=Ø THEN 645
48Ø KO=Ø
485 REM FUER UMWANDLUNG VON
CTRL
49Ø FOR J=1 TO L
495 REM - - - - - - - - - - 
 - - - - - - - - - - - - - -
5ØØ REM ISOLIERUNG DES GESUC
HTEN CTRL-ZEICHEN
5Ø5 IS$=SEG$(A$(Z),J+KO,1)
51Ø REM '1' IN SEG$, WEIL CT
RL IMMER NUR EIN ZEICHEN
515 CODE=ASC(IS$)
52Ø IF CODE<=128 THEN 64Ø
525 IF CODE<147 THEN 545
53Ø REM  EIN NICHT VEREINBAR
TES CTRL-ZEICHEN GEWAEHLT
535 IS$=" "
54Ø GOTO 64Ø
545 REM CODE ERGIBT EIN DEFI
NIERTES CTRL-ZEICHEN
55Ø REM - - - - - - - - - - 
 - - - - - - - - - - - - - -
555 L$=SEG$(A$(Z),1,J-1+KO)
56Ø R$=SEG$(A$(Z),J+1+KO,L-J
)
565 REM -*-*-*-*-*-*-*-*-*-*
-*-*-*-*-*-*-*-*-*-*-*-*-*-*
57Ø REM WIRD IS$ EIN, ZWEI O
DER DREI ZEICHEN LANG      *
575 IF CODE=129 THEN 615
58Ø REM ES LIEGT ESC-SEQUENZ
 VOR. ZWEI OD.DREI ZEICHEN *
585 KO=KO+1
59Ø REM ESC-SEQUENZ MINDESTE
N ZWEI ZEICHEN LANG        *
595 IF CODE<139 THEN 615
6ØØ REM ESC-SEQUENZ DREI ZEI
CHEN LANG                  *
6Ø5 KO=KO+1
61Ø REM WERT VON KO WIRD DUR
CH ESC LAUFEND ERHOEHT     *
615 NR=CODE-128
62Ø REM -*-*-*-*-*-*-*-*-*-*
-*-*-*-*-*-*-*-*-*-*-*-*-*-*
625 IS$=CODE$(NR)
63Ø A$(Z)=L$&IS$&R$
635 REM ZUSAMMENBAU DER ZEIC
HENKETTE WIEDER ERFOLGT
64Ø NEXT J
645 NEXT Z
65Ø REM -------------------
----------------------------
655 REM DRUCK DES UMGESCHLUE
SSELTEN TEXTES = SCHRITT 4
66Ø REM -------------------
----------------------------
665 CALL CLEAR
67Ø PRINT "   TEXT ZUR AUSGA
BE BEREIT"
675 PRINT
68Ø PRINT "      DRUCKER VORB
EREITEN"
685 PRINT
69Ø PRINT "              DANN
"
695 PRINT
7ØØ PRINT "      TASTE 'D' DR
UECKEN"
7Ø5 CALL KEY(Ø,T,ST)
71Ø IF T>68 THEN 7Ø5
715 IF T<68 THEN 7Ø5
72Ø I=Z-1
725 CALL CLEAR
73Ø PRINT "   ES WIRD JETZT
GEDRUCKT"
735 OPEN #1:"RS232.BA=48ØØ.D
A=8",OUTPUT
74Ø REM OPEN-ANWEISUNG DEM D
RUCKER ANPASSEN
745 FOR Z=1 TO I
75Ø PRINT #1:A$(Z)
755 NEXT Z
76Ø CLOSE #1
765 END
```

oft auch als Killercode bezeichnet – muß in gleicher Weise an den Drukker mittels PRINT-Anweisung gesendet werden. Hersteller von Matrixdruckern wählen hierfür drei verschiedene Varianten:

- Die Aufhebung erfolgt durch erneutes Senden des gleichen ESC-Sequenz.
- Die Aufhebung geschieht durch die gleiche Fluchtsequenz. Lediglich der Parameter n ist jetzt nicht 1 (entspricht einschalten) sondern 0 (entspricht ausschalten). Typisches Beispiel ist der Unterstreichungsmodus.
  Unterstreichung ein:
  PRINT #1:CHR$(27)&CHR$(45)&CHR$(1)
  Unterstreichung aus:
  PRINT #1:CHR$(27)&CHR$(45)&CHR$(0)
- Eine Aufhebung wird durch eine andersartige Sequenz bewirkt, z.B. Doppeldruck ein (ESC G), Doppeldruck aus (ESC H).

DIREKTES EINBINDEN EINER FLUCHTSEQUENZ WÄHREND DES SCHREIBENS VON TEXT

Der Informationswert von Bedienungsanleitungen für Matrixdrucker hat sich in letzter Zeit sichtbar verbessert. Bei den genannten drei Druckern ist der Aufbau der Anleitung in methodischer und didaktischer Hinsicht sogar ganz hervorragend. Bei Kenntnis des vorher genannten Sachverhalts werden deshalb kaum Schwierigkeiten beim direkten Einbinden einer Fluchtsequenz innerhalb eines TI 99/4A-Programms auftreten. Ein solches Einbinden ist für bestimmte Ansteuerungsmöglichkeiten nicht nur empfehlenswert sondern manchmal unumgänglich (siehe hierzu auch Tölke, Textverarbeitung mit Programmen für TI 99/4A und VC 20. Vieweg Verlag 1984, S. 37 Beispiel 13).

Eine ganze Reihe von Ansteuerungsmöglichkeiten möchte man aber gerne direkt vom eingegebenen Text aus bewirken. Hierzu gehört beispielsweise das Unterstreichen eines Worts und vieles mehr. Leider geht meines Wissens bisher keine Anleitung auf diese Problematik ein.

Bei der Lösung dieses interessanten Problems muß die Schwierigkeit umgangen werden, daß es auf direktem Weg nicht möglich ist, eine ESC-Sequenz in den Text beim Schreiben einzubinden. Aus Platzgründen wollen wir uns auf eine einzige recht interessante Lösungsvariante beschränken. Die nachfolgenden Ausführungen sollen dabei durchaus beispielhaften Charakter besitzen. Daran erkennbar, daß nur eine Untermenge der vorhandenen Ansteuerungsmöglichkeiten der Drucker MX-80, RX-80 und FX-80 ausgeschöpft werden. Ganz offensichtlich steigt von Modell zu Modell die Vielfalt des Druckkomforts und das bei konstanten Preisen, bezogen immer auf das aktuelle neue Modell.

Modell	MX-80	RX-80	FX-80
Ansteuerungsmöglichkeiten gesamt	40	52	71
davon ESC-Sequenzen	27	40	56

Das noch schrittweise zu erläuternde Programmbeispiel läuft direkt mit den hier genannten Druckern, wobei es nur mit dem MX-80 praktisch getestet wurde. Eine Umsetzung auf andere Druckertypen oder eine andersartige Auswahl der ESC-Sequenzen wird nicht schwer fallen, wenn man das angewandte Prinzip beachtet.

Für die Realisierung unseres Vorhabens sind „Steuercodetasten" notwendig, die folgende Bedinungen erfüllen müssen.

- ☐ Möglichkeit des Verzichts auf darstellbare Zeichen auf dem Bildschirm bei Betätigen der betreffenden Taste (Tastenkombination), wenn Text geschrieben wird, die sich aber trotzdem einbinden lassen.
- ☐ Mittels ASCII-Wertfunktion (ASC) muß die gedrückte Taste (Tastenkombination) einen Code liefern.
- ☐ Der zugeordnete Code muß einen bestimmten numerischen Bereich umfassen und möglichst ohne Unterbrechung fortlaufend sein.
- ☐ Nach Drücken dieser Taste (Tastenkombination) muß dieses innerhalb der Zeichenkette einen Platz einnehmen, auch wenn dieser auf dem Bildschirm nur als scheinbares Leerzeichen indirekt sichtbar wird. Dies erleichtert eine später eventuell notwendige Korrektur ganz erheblich.

Diese Bedingung erfüllt die Tastenkombination CTRL in Verbindung mit einem Zeichen (im weiteren nur noch CTRL-Zeichen genannt) in geradezu idealer Weise. Auch wenn dafür laut Bedienungsanweisung für viele Zeichen schon eine Funktion vergeben ist. Bevor diese definierte Wirkung aber eintreten kann, wird im Programm der eigentliche Code (z.B. CTRL A besitzt den Code 129) um 128 vermindert und liefert jetzt den numerischen Wert 1. Bezogen allein auf alle Buchstaben des Alphabets entstehen so die Werte 1 bis 26 (CTRL Z). Die jeweilig so erzeugte Zahl wird als Wert der Varianten 'NR' zugewiesen und erreicht so den String mit der eigentlichen Steuerwirkung (CODE$(NR)) über den Index jederzeit.

wird fortgesetzt

Assembler leicht gemacht

Beim letzten Mal mußte der kleine Tip für die Experten entfallen. Das waren dann die berühmten 10 Zeilen zuviel beim Umbruch, d.h. 10 Zeilen wären über die Seite hinausgegangen und „Gummi-Seiten" gibt es leider noch nicht.

Deshalb diesen Tip nun hier am Anfang: Bei der Dateibehandlung unterscheidet der TI 99/4A ja 5 verschiedene Formate: Program, DIS/VAR, DIS/FIX, INT/VAR und INT/FIX. Gemäß dem Handbuch zum Editor-Assembler kann über die Abfrage des Status einer Datei auch das Format in Erfahrung gebracht werden. Dazu wird einfach ganz normal die Datei über DSRLNK angesprochen. Der OpCode im PAB ist dafür dann >09. Hier ist in den meisten Disk-Controlern ein Softwarefehler enthalten. DIS/FIX-Dateien werden bei der Statusabfrage als DIS/VAR zurückgegeben.

Nun wollen wir aber weiter in unserem Assemblerkurs fortfahren. Viele Anfragen erhielt ich nach der Belegung des VDP-Ram's, d.h. was steht dort wo. Dies soll dann auch heute unser Thema sein. Vorab muß aber dazu gesagt werden, daß sich die Belegung je nach Modul unterscheidet. Wir wollen uns heute auf TI-Basic, Extended Basic und das Editor/Assembler-Modul unter der Funktion LOAD AND RUN beschränken, dabei ist das Mini-Memory unter LOAD AND RUN gleich wie das Assembler-Modul.

Rein für die Bildschirmdarstellung braucht der Video-Display Prozes-

```
*****************************
* VDP-REGISTER UND TABELLEN LADEN
*

SETREG LI   1,REG          * REGISTER WERTE
REG1   MOV  *1+,Ø          * WERT AUS DER TABELLE
       JEQ  REGEND
       BLWP §VWTR          * REGISTER LADEN
       JMP  REG1
REGEND MOVB §REG+1,§>83D4 * KOPIE VDP-REG 1 FUER INTERRUPT
       LI   Ø,>Ø3ØØ        * >DØ AUF BEGINN DER SPRITE
       LI   1,>DØØØ        * ATTRIBUTE TABLE, KEINE SPRITES
       BLWP §VSBW          * WERDEN ABGEBILDET
       LI   Ø,>Ø38Ø        * COLOR TABLE
       LI   1,>FØØØ        * VORDERGRUND WEIS AUF TRANSPARENT
REG2   BLWP §VSBW
       INC  Ø
       CI   1,>Ø3AØ
       JNE  REG2
       LI   1,>Ø9ØØ        * BEGINN AB CHARAKTER >2Ø = SPACE
       MOV  1,§>834A
       BLWP §GPLLNK        * GR. BUCHSTABEN LADEN
       DATA >ØØ18
       BLWP §GPLLNK        * KL. BUCHSTABEN LADEN
       DATA >ØØ4A
       LI   Ø,>Ø8FØ        * PATTERN DESCRIPTOR TABLE
       LI   1,CURSOR       * CURSOR CHAR
       LI   2,>ØØ1Ø        * 16 BYTES
       BLWP §VMBW
       RT

CURSOR DATA >3C42,>99A1,>A199,>423C * COPYRIGHT ZEICHEN
       DATA >3C3C,>2424,>2424,>3C3C * CURSOR

* DATEN FUER DIE VDP-REGISTER


REG    DATA >Ø1EØ  * REG. 1: 16KBYTE, GRAPHICS MODE
       DATA >Ø2ØØ  * REG. 2: BASIS SCREEN IMAGE TABLE
*                  * WERT MAL >4ØØ (HEX!)
       DATA >Ø3ØE  * REG. 3: BASIS COLOR TABLE
*                  * WERT MAL >4Ø (HEX!) >4Øx>ØE=>38Ø
       DATA >Ø4Ø1  * REG. 4: BASIS PATTERN DESCRIPTOR TABLE
*                  * WERT MAL >8ØØ (HEX) >8ØØx>Ø1=>8ØØ
       DATA >Ø5Ø6  * REG. 5: BASIS SPRITE ATTRIBUTE LIST
*                  * WERT MAL >8Ø (>8Øx>Ø6=>3ØØ)
       DATA >Ø6ØØ  * REG. 6: SPRITE DESCRIPTOR TABLE
*                  * WERT MAL >8ØØ
       DATA >Ø7F4  * REG. 7: FARBE FUER TEXTMODE GRAU
*                  * UND HINTERGRUND DUNKELBLAU
       DATA >ØØØØ  * KENNWERT ENDE DER TABELLE
```

sor einige Tafeln. Im einzelnen sind dies:
1. Screen Image Table, auch als Bildschirmtabelle zu bezeichnen. In diesem Bereich stehen die abzubildenden Zeichen in ASCII. Dabei ist die linke obere Ecke des Bildschirms der Anfang der Tafel, der Buchstabe rechts daneben das nächste Byte usw. Das 33. Byte ist also das erste Zeichen in der zweiten Zeile. Insgesamt benötigt diese Pattern Descriptor Table 768 Bytes (24 Zeilen a 32 Spalten).
2. Die Color Table,also die Farbtabelle. Diese bestimmt die Farben der Zeichen jedes Charaktersatzes (jeweils 8 Zeichen, wie in CALL

COLOR im Basic, nur stimmen die Nummern der Zeichensätze nicht überein, da der Video-Display-Prozessor von 0 für die ASCII-Character 0 – 7 rechnet). Für jeden Charaktersatz brauchen wir hier ein Byte, insgesamt also 32 Bytes (256 geteilt durch 8). In jedem Byte bestimmen die ersten vier Bits (jeweils vier Bits werden auch als Nybble bezeichnet, hier ist also das höchstwertige Nybble gemeint), die Vordergrundfarbe und das niedrigwertige Nybble die Hintergrundfarbe. Auch hier gelten wieder die Werte aus dem Basic für die Farben, nur müssen wir 1 davon abziehen, also ergibt sich 0 für Transparent und >F bzw. 15 für Weiß.

3. Die Pattern Descriptor Table, die Tafel zur Bestimmung der Zeichen. Diese wird aus dem Basic mit CALL CHAR geladen und ist eigentlich für jedes Zeichen genauso aufgebaut wie der 16stellige String, den wir in Basic eingeben. Dieser String kann ja auch als 8Bytes Hexadezimal gesehen werden. Diese Tabelle umfaßt also 256 mal 8 Bytes, macht 2048 Bytes. Um Platz zu sparen, werden aber im Basic nicht alle 256 Zeichen die der Video-Prozessor darstellen kann ausgenutzt, so daß die Tabelle dann kleiner gehalten ist.

4. Die Sprite Attribute Tabelle, also die Sprite Werte Tabelle. Hier befinden sich jeweils 4 Bytes für jeden Sprite. Das erste bezeichnet die Lage in vertikaler Richtung (>FF ist ganz oben, dann folgt >00 bis >BE nach unten) und das zweite Byte bestimmt die Lage in horizontaler Richtung (>00 ist ganz links, >FF rechts). Das nächste Byte legt den Charakter-Code fest und das letzte Byte bestimmt die Farbe des Sprite im niedrigwertigen Nybble. Vom höherwertigen Nybble wird nur das letzte Bit verwendet. Ist es nicht gesetzt, ist die obere linke Ecke des Sprites mit den Angaben für die Position bestimmt und der Sprite wandert auf der rechten Seite des Bildschirmes sauber heraus. Ist dieses Bit gesetzt, dann ist die rechte obere Ecke bestimmt, d.h. der Sprite wird um 32 Pixel nach links verschoben und wandert auf der linken Seite richtig heraus. Der Video-Prozessor kann 32 Sprites verwalten, also ist diese Tabelle insgesamt 128 Bytes lang, aber auch hier wird im Extended Basic etwas Platz gespart und nur 28 Sprites werden verwendet.

5. Die Sprite Motion Table, Sprite Bewegungs-Tabelle. Diese Tabelle wird eigentlich nicht vom Video-Prozessor benötigt, sondern wird ausschließlich von der Interrupt-Routine verwaltet. Je Sprite sind auch hier wieder 4 Bytes vorhanden. Die ersten beiden geben den Geschwindigkeitswert in vertikaler und horizontaler Richtung an. Negative Zahlen geben dabei eine umgekehrte Bewegung. Die beiden weiteren Bytes werden nur von der Interrupt-Routine zur zwischenzeitlichen Speicherung von Werten benutzt.

6. Die Sprite Descriptor Table. Diese Tabelle ist genauso wie die Pattern Descriptor Table aufgebaut, nur gilt sie eben für die Sprites. Häufig wird sie auch mit der Pattern Descriptor Table zusammengelegt, so auch in Extended Basic. So brauchen wir darauf hier nicht noch weiter einzugehen.

Was liegt nun von diesen Tabellen wo im VDP-RAM. Zuerst unter TI-Basic:

VDP-Adresse	Tabellenbezeichnung
HEX DEZ	
>0000 0	
	Screen Image Table
>02FF 767	
>0300 768	
	Color Table
>031F 799	
>0320 800	
>03EF 1007	
	Pattern Descriptor Table mit Offset 96 (HEX> 60) Start daher eigentlich ab> 0000
>03F0 1008	Start mit Charakter für ASCII 30
>07FF 2047	Ende variable je nach definierten Zeichen

Der Bereich von> 0320 bis> 03BC wird vom TI-Basic zum Umwandeln von Eingabezeilen in das Basic-Format (Crunchen) verwendet.

Nun zum Extended Basic:

VDP-Adresse	Tabellen-Bezeichnung
HEX DEZ	
>0000 0	
	Screen Image Table
>02FF 767	
>0300 768	
	Sprite Attribut Table
>036F 879	
>0370 880	
>03EF 1007	
	Pattern Descriptor Table mit Offset 96 (HEX> 60) Start daher eigentlich ab> 0000
>03F0 1008	Start mit Charakter für ASCII 30
>077F 1919	
>0780 1920	
	Sprite Motion Table
>07FF 2047	
>0800 2048	
	Color Table
>081F 2079	

Der Bereich von >0370 bis >03EF und> 08C0 und >0967 wird vom Extended Basic zur Zwischenspeicherung von Werten und zum Umwandeln von Eingabezeilen benutzt. Das Editor-Assembler-Modul nimmt weniger Rücksicht, braucht es ja auch gemäß seiner Bestimmung nicht. Es ergibt sich folgende Aufteilung:

VDP-Adresse	Tabellen-Bezeichnung
HEX DEZ	
>0000 0	
	Screen Image Table
>02FF 767	
>0300 768	
	Sprite Attribut Table
>037F 895	
>0380 896	
	Color Table >039F 927
>03A0 928	
>03FF 1023	
>0400 1024	
	Sprite Descriptor Table nur Zeichen ab> 80 (128 dez.) bis> EF (dez 239) benutzbar.
>077F 1919	
>0780 1920	
	Sprite Motion Table
>07FF 2047	
>0800 2048	
	Pattern Descriptor Table ohne Offset für 256 Charakter
>0FFF 4095	
>1000 4096	

Im Basic befinden sich nun oberhalb dieser Tabellen im VDP-RAM das Programm, die Werte aller Variablen und noch einige „Kleinigkeiten“. Beim Editor-Assembler ist der Platz frei. Bei einem angeschlossenen Disk-Laufwerk ist ganz am oberen Ende des VDP-RAM's noch ein Speicher reserviert, in dem nichts verändert werden sollte. Zu diesem Speicherbereich werden wir, wie auch zum Aufbau eines Basicprogrammes, in einem anderen Beitrag kommen. Heute soll am Schluß ein kurzes Beispielprogramm stehen, wie alle notwendigen Tafeln des VDP-RAM's und die Register des Video-Prozessors geladen werden können. Die Kommentare geben auch Aufschluß über die Bedeutung der VDP-Register. *Heiner Martin*

Haste Töne? Und was für welche!

>8400 oder dezimal -31744 ist die Adresse des in der TI 99-Konsole integrierten Soundchips. Immer wenn wir den Befehl CALL SOUND aufrufen, werden die einzelnen Parameter hintereinander auf den 8 Bit breiten Eingang des Chips gelegt – Grund genug, einmal genauer auf die Reihenfolge und Art der Sounddaten einzugehen. Um das nun folgende nachzuvollziehen, sind keine allzu großen Hardwarevoraussetzungen notwendig: es genügt, im minimalen Falle, das Ex-Basicmodul und die Miniassembler-Erweiterung von Radix. Natürlich geht's auch mit 32 k oder Minimem – halt irgendwas, was den Befehl CALL LOAD auszuführen vermag . . .

Für den TI gibt es zwei verschiedene Möglichkeiten, den Tongenerator – oder besser gesagt die Generatoren – anzusteuern. Erster Weg führt über eine Parameterliste, die irgendwo im (VDP)RAM steht und über GPL oder eine Assemblerroutine auf den Generator gejagt wird. Der zweite Weg ist die direkte Ansteuerung des Soundchips – die Werte werden also im Direktmodus oder im Programm auf den Chip gegeben. Wir interessieren uns hier für die zweite Methode – immer in dem Bestreben, über ein paar „Pokes" eine Menge zu regeln . . .



Die grundliegende Einschränkung bei direkter Ansteuerung des Chips ist, daß wir keine Tondauer – entspricht dem ersten Wert in Call Sound – eingeben können. Die wird beim TI nämlich „über Software" erzeugt und somit hat der Soundchip kein Register für die Zeit, die ein bestimmter Klang andauern soll. Dazu noch eine Anmerkung: im Call Sound Statement kann ein Ton laut TI-Handbuch zwischen 1ms und 4250 ms dauern. Intern – auf GPL-Ebene – wird dieser Wert allerdings zunächst mit 6 multipliziert und dann durch 100 dividiert. Somit wird aber aus der ursprünglichen 2 Byte-Zahl eine 1 Byte-Integerzahl. Wirkung: Egal ob wir die Zeitdauer mit 1ms oder mit 20ms angeben, es ist dasselbe!
Noch eine Bemerkung zum Handbuch und Call Sound. Dort steht nämlich, daß die Werte für die drei Tongeneratoren Integerzahlen sein müssen. Stimmt aber nicht, man kann nämlich genausogut Fließkommazahlen eingeben. – Nicht nur, daß man sich dadurch während des Programmablaufes Speicherplatz und Zeit spart – weil man etwa auf die INT-Funktion verzichten kann –: bei niedrigen Frequenzwerten „nimmt" der Tonprozessor die Fließkommazahlen auch an. Gebt doch mal spaßeshalber nacheinander die Werte 110.1...... 110.9 auf den ersten Tongenerator. Wer ein gutes Ohr hat, vernimmt deutlich Frequenz(=Tonhöhe)-unterschiede. In der Praxis ist das nicht ganz unwichtig: viele musikalisch untermalte Programme klingen deutlich „schräg", weil die Frequenzwerte als Integerzahlen eingegeben werden. Besser ist die Anwendung des Algorithmus aus dem englischsprachigen Ex-Basic-Handbuch

---

SPART JEDE MENGE SPEICHERPLATZ

– Seite 171 – da hier die genauen Tonverhältnisse berechnet werden. Im übrigen spart die Verwendung dieses Algorithmus gegenüber endlosen Datazeilen auch jede Menge Speicherplatz. Kommen wir aber, wie versprochen, zur direkten Ansteuerung der Generatoren. Wie schon oben angedeutet, gibt es hierbei den Parameter „Dauer" nicht mehr. Einmal programmiert und es tönt ewig. Das muß nicht unbedingt von Nachteil sein, da viele Spielprogramme erst durch andauernde Geräuschuntermalung richtig schön werden – wir ersparen uns also das Anspringen einer Soundroutine mitten in der Joystickabfrage und das kann u.U. so manchem Basicspiel geschwindigkeitsmäßig auf die Beine helfen. Noch eine Eigenart: die programmierte Frequenz – oder das programmierte Rauschen – bleiben solange erhalten, bis wir einen neuen Wert auf das entsprechende Register geben. Mehr dazu später. Damit es auch gleich mächtig lärmt, fangen wir direkt mit dem an, was die Angelsachsen gerechterweise als „noise" bezeichnen: der Rauschgenerator. Eingeschaltet wird dieser mit CALL LOAD(-31744, A), wobei A zwischen 224 – entspricht -1 bei Call Sound – und 231 – entspricht -8 – betragen darf. Nach dem ersten "Poke" werden wir noch nichts hören – die Soundgenera-

toren sind noch abgeschaltet. Dieser Tatsache können wir mit CALL LOAD (-31744,B) abhelfen; B ist ein Wert zwischen 240 und 255 – mit 255 wird der Rauschgenerator abgeschaltet, 240.....254 entspricht den einzelnen Lautstärkestufen in abfallender Reihenfolge. CALL LOAD (-31744,227)::CALL LOAD (-31744,240) entspricht also einem CALL SOUND(X,-4,0) – wobei wir nicht vergessen dürfen, daß der Parameter „Dauer“ beim poken ja entfällt. Die „interessantesten“ Geräusche, die man mit dem TI produzieren kann, dürften wohl die sogenannten modulierten Rauschtöne sein. Wie es im TI-Basic-Handbuch steht, kann man die Rauschwerte -4 und -8 mit der Frequenz des Tongenerators Nr. 3 modulieren. Hier wird deutlich, daß gerade auf die TI-Generatoren der englische Begriff „noise“ besser als das deutsche Rauschen paßt: -4 ist nämlich eher eine Art „Knattern“ oder technisch gesehen ein tiefer Rechteckton. Moduliert man diesen Klang,

SEHR TIEFE TÖNE

so lassen sich sehr tiefe Töne erzeugen: im Basic über CALL SOUND (4250,110,30,110,30, 110,30,-4,0) oder über LOAD mit den hintereinander eingegebenen Werten 227,240,207,63,223. Jetzt wird allerdings deutlich, daß CALL SOUND wesentlich einfacher ist. Man braucht beim Poken ja immerhin 5 Befehle... Was viele Ex-Basic-Benutzer nicht wissen: man kann mit einem einzigen CALL LOAD mehrere Werte hintereinander auf dieselbe – oder auch auf verschiedene – Speicherstellen geben! Die Syntax:
CALL LOAD(ADR1, BYTE1,“ “,ADR2, BYTE2, “ “,.......,ADRn, BYTEn) wobei maximal 15 Speicherstellen angepoked werden können.

“Reset“ für das Modul “CONNECT FOUR”

Hier ein kleiner Tip, der nicht im Begleitheft zu diesem Modul steht. Es passiert schon mal, daß man sich in den Menüs dieses Spieles verwählt. Man muß dann dieses Spiel aber nicht zu Ende spielen oder gar –Fctn Quit – drücken, sondern einfach – Shift 2 –. Dies bewirkt, daß man zum Demospiel zurückkehrt. Dies geht auch, wenn man schon im Spiel ist.

Andreas Zeller

Die beiden Gänsefüßchen kennzeichnen einen Leerstring – man kann anstatt “ ” auch A$ reinpakken, falls gewährleistet wird, daß diese Variable ein Leerstring bleibt. Diese wenig bekannte Tatsache gilt übrigens für alle TI-Module, die Call Load kennen, beim Editor/Assembler kann man auch mit CALL PEEK so verfahren – das geht beim Ex-Basic leider nicht. Schneller und speicherplatzsparender geht's also mit CALL LOAD(-31744, 227, “ ”,-31744,240,“ ”, -31744,207, “ ”,-31744, 63,“ ”,-31744,223). Die letzten drei Bytes steuern den Tongenerator Nr. 3. Die Ansteuerung ist ein wenig kompliziert:
1. Byte von 192 bis 207; enthält die 4 niederwertigsten Bits für Tonhöhe
2. Byte von 0 bis 63: enthält die 6 höchstwertigsten Bits für Tonhöhe
3. Byte von 208 bis 223; enthält Lautstärke – wie beim Rauschgenerator.
Da die anderen beiden Tongeneratoren prinzipiell genauso angesteuert werden, hier die Werte:
Tongenerator Nr. 2
1. Byte von 160 bis 175
2. Byte von 0 bis 63
3. Byte von 176 bis 191
Tongenerator Nr. 1
1. Byte von 128 bis 143
2. Byte von 0 bis 63
e. Byte von 144 bis 159.

Um das Ganze zu verdeutlichen, müssen wir zunächst wissen, wie man die aus CALL SOUND bekannten Frequenzwerte in die nötigen ein-Byte-Zahlen umrechnet... mit folgender Subrotuine nämlich:

```
1000 REM***Frequenz
     zu Poke***
1010 F=INT(111860.8/
     F+0.5)
1020 MSB=INT(F/16)
1030 LSB=F-(MSB*16)
1040 RETURN
```

Um jetzt beispielsweise dem Tongenerator Nr. 3 verschiedene Töne mit aus CALL SOUND-Zeiten bekannten Frequenzwerten zu entlocken, müssen wir noch folgendes eintippen:

```
100 ACCEPT AT(10,1)
    ERASE ALL:F
110 GOSUB 1000
120 CALL LOAD(-31744,
    192+LSB,“ ”,-31744,
    MSB,“ ”,-31744,208)
130 GOTO 100
```

Wollen wir damit auch noch ein Rauschen modulieren, so genügt es, am Anfang des Programmes die Zeile
90 CALL LOAD(-31744, 227,“ ”,-31744,240) einzufügen. Wollen wir nur das modulierte Rauschen und nicht dagegen den Ton von Generator Nr. 3 hören, so wird der letzte Wert in Zeile 120 zu 223. Wenn wir das gerade abgetippte Programm in einem „Benchmark“ gegen eines mit CALL SOUND laufen lassen würden, so sähe es ganz schön schlecht für die Poke-Methode aus...

VARIABLE EINSETZEN!

Erste Programmiererpflicht in diesem Falle: Variablen anstatt Konstanten einsetzen. Also etwa Z$ anstatt “ ” und Z anstatt -31744. Bitte die Variablennamen möglichst kurz halten, das spart Speicherplatz und Zeit. Außerdem nach Möglichkeit keine Arrays verwenden, die laufen auf dem TI nämlich langsamer als einfache Variable – bei anderen Computern ist es in der Regel umgekehrt!

Jetzt liegt man über LOAD in einem leichten Vorteil gegenüber SOUND, kann allerdings den berechtigten Gedanken äußern, daß sich für so einen geringen Geschwindigkeitsvorteil das ganze Nachdenken und Programmieren wohl kaum gelohnt haben kann... über LOAD kann man aber so einige Sächelchen tätigen, die mit CALL SOUND nicht möglich sind, alsda wären:
Aus- und Einschalten einzelner Generatoren
Erzeugen einer Hüllkurve für jeden einzelnen Generator
Unabhängige Verwendung der einzelnen Generatoren

VARIABLE KURZ HALTEN

Beispiel für eine Hüllkurve:
CALL LOAD(-31744,224, “ ”,-31744,240,“ ”, -31744,241,“ ”,.........“ ”, -31744,255)
Erklärung des Ganzen: alle ‚-31744's sind natürlich die Adressen; der Wert ‚224‘ initialisiert den Rauschgenerator – entspricht dem Basic-Wert ‚-1‘ – und alle Zahlen von ‚240‘ bis ‚255‘ regeln die Lautstärke kontinuierlich herunter. In diesem Befehlsformat geht das wesentlich gleichmäßiger vor sich, als wenn wir es über sukzessive CALL SOUND oder -LOAD-Befehle versuchen. Wenn wir Variablen anstatt der Festwerte einsetzen, so geht's noch etwas schneller und man kriegt mehr Befehle in eine Zeile – aber: Variablennamen bitte möglichst kurz halten, also A,B,C anstatt TON1, TON2, TON3 – you know what I mean. Natürlich kann man auch verschiedene Werte für verschiedene Generatoren und ihre Lautstärken eingeben – aber schafft Euch fürs Experimentieren besser einen Kopfhörer an, wenn Ihr nicht plötzlich den Tierschutzbund vor der Türe haben wollt.

Jürgen Heinzerling

BÖRSE

Suche: 32K-Ext. durchgef. Bus + Centron, CBM 64/ 128 + Floppy, Spitzen-Biorh.- u. Astrolog. Prog. mit Hardcopy, Druckerbuffer f. GP 550, etc. – BIETE: TI-Centr. In+f., MBI: 150,– DM Meschkat, Sandweg 10, 3500 Kassel, Tel. 0561/880100 abends

---

Verk.: TI 99/4A, Ex-Basic, Box, 32K-Ram, RS 232, Dsik.-Contr. 1 Laufwerk. Daten v. Textv. Literatur. VB 2800, 0231/87 15 81

Verk. Seikosha GP50 m. Schnittstelle für 994A + Kabel 500,– Tel.: 02101/ 46 49 68 bei Weiczorek

---

Endlich!!! 3 neue Spiele aus dem Hause SOLO-SOFT. Informationen in dieser Ausgabe (s.u.)

TI-99 Tips und Tricks (Ein Data Becker Buch) für 30,– DM + Nachnahme zu verk. Joachim Bregulla, Ostseestr.5, 435 Recklinghausen Tel. 02361/46219

32K-Erweiterung für TI99, extern batteriegepuffert und durchgeführter Bus für 245 DM dazu passende Assemblersoftware Tel. 0241/86447

M.P.J.M.P.J.M.P.J.M.P.J. M.P.J. ist ein Strategiespeil Super 10 DM Ein mußt für TI'ler. Info: 0221/697254

Wer hilft 99/4A-Fan i.d. DDR? Gesucht: billiges 32K RAM, Konsole z. ausschlachten, Schaltplan u. Bauanltg. f. Hardware, def. Hardware jed. Art. Kontaktadresse: Otto 42 OB 1, Feldmannstr. 54, Tel. (ab 19 h) 0208/860892

Crazy Man, Flugsimulator, Frogger u.ä. super Listings zu Superpreisen!!! INFO gegen Rückporto bei: Selim Özdogan, Wuppertaler Str. 26, 5000 Köln 80

SUCHE MÖGLICHST BILLIG: Return to the Pirates Isle, Speechsynth., TEII, Minimem + SW, Othello, orig. TI Joysticks Angebot bei: Selim Özdogan, Wuppertaler Str. 26, 5000 Köln 80

---

für E/A, MM u. XBasic + 32 K. Info gg. Freiumschlag. Sofort lieferbar! Mathias Eichhorn, Ziegelhech 1, 6240 Königstein 4

TI99/4A+Lehrg.+100Progr. 299 DM/16 Module ab 15 DM/TI-Cass. 15 DM 4TI-Bücher ab 10DM/TI+ Ex. Progr. auch Tausch/ Info 1 DM bei Ralf Schmitz, Am Telegraf 14, 5068 Odenthal 3

Suche Sprachmodul Anruf bei G. Heinrichs Tel. 06591/7303

DRUCKE Ihre Prg. für 6.– DM je Listing. Entferne Listschutz für 3.– DM pro Gassette! Info 50 Pfg. anfordern oder Geld an: Matthias Orf, Birkenallee 34, D-3507 Baunatal 1, Tel. 0561/497990

Drucke f. nur 3.– DM Listings TI-X-Bas. MM (Kassette). Suche günstig TI99/4A (auch defekt), Module jeder Art-X-Bas. Modul und E/A-Handbuch, auch Kopie. Ulrich Hustedt, Schippenbeiler Weg 5, 3070 Nienburg 2

Suche: PeriBox, Disk-Gontr., Laufwerk, 32 K Tel. 07231/74142 18-20 Uhr

---

High-Res.-Grafik + 32 Sprites für Ti99/4A + Minimen. 16320 Pkte. 27 neue LINK-Befehle Kassette + Beschr. DM 30,– (Scheck/ bar) Dr. M. Kausch, Semmelweisstr. 151, 5000 Köln 80

TII 99/4A + EX-Basic + Joystmodüle-PAGMAN – D. Kong + Rec. Kabel + Programme auf Kassette + Bücher 530 DM Tel. 08161/83587

TI 99/4A + Ext. B. + Minimen + Joysticks + 2 Rec. Kabel + Parsec + Munchman + viel Software + viel Literatur VB DM 700,– Tel. 040/7547576 Jürgen Gerkens

---

Verk: TI99/4a + Ex Basic 11 + Gass. Rec. + Modul gegen Höchstgebot. F. Simonis, Lev. 3, Tel. 02171/55305

Verkaufe MiniMemory 100,– DM + andere Module spottbillig! NN+5DM, x Liste anfordern!x Jürgen Kohn, Stettener Str. 41, 7450 Hechingen, T. 07471/16820

Su. ext. Bas. Modul möglich m. dt. Anleit. sow. Microsurgeon. Bin an kontakten zwckes Soft und Harware inters. Robert Kirchmayer Sollerstr. 24, 8110 Murnau

---

Tausche mein Tombstone City Modul gegen Invaders. Modul. Andreas Scholz, Londonerring 6, 6700 Ludwigshafen

---

TI 99/4A + Ext. B. + V. Chess. + Defnd. + Kabel + 6 Cass. + Fast alle Bücher + 30 TI zeitgen + Listings + Schaltpl., alles kaum benutzt 350 DM, Evtl. S/W Mon. + Cassr. + Joys. = 450 DM. T. 040/7207870

Ich verkaufe folgende TI-Module

Editor/Assembler	110,–
Disk Manager 2	90,–
Pic Man	60,–
Pers. Record Keeping	50,–
Statistik	40,–
Indoor Soccer	30,–
TI Invaders	30,–
Parsec	35,–
M*A*S*H	40,–
Burger Time	40,–
Alien Addition	15,–
A-maz-ing	25,–
Othello	25,–
Buck Rogers	40,–
Star Trek	40,–

Tel. 0212/43140

Habt Ihr Lust, Eure Programmsammlung zu vergrößern? Hättet ihr Lust zum Tauschen? Meldet Euch doch einfach bei mir! Tel. 07156/34941. Am Ende der anderen Leitung meldet sich dann Garlos Jarque. Ich freue mich schon auf Eure Anrufe!

Suche das Modul Pole Position für den TI möglichst billig! Angebote an Carlos Jarque Ludwigsburgerstr. 14, 7257 Ditzingen 1 oder Tel. 07156/34941

Sonderangebot: TI99/4A + ext. Diskcontroller, + Doppeldiskstation, + Druckerinterface nur DM 850,–. Nur bei Michael Franz Elektronik, Hebbelweg 17, 2082 Tornesch Tel. 04122/54616

Super Software? Mini Preis?? NA LOGO! Von Adventures bis zum Flugsimulator alles da!!! Wegen Systemaufgabe 20 Pgms. nur 19 (!) Märker! 50 Pgms. nur 45 (!) Kröten! usw. ... Wo denn nur, häh? Klar, bei dieser Nr.: 02541/41531!!

Suche Disk-Manager 2 neu oder gebraucht. Tel. 05103/3020 (ab 18 Uhr)

Verkaufe orig. TI-Disklaufwerk für P-Box (neu) Preis: 380,– DM (Tausche gegen ext. Laufw. mit Gontrr. möglich) Tel: 07144-5227 18.00

Hey Freaks! 32KRAM-Speicher exh.: BS: 195,– DM o. FG: 240 DM. Info von H. Ayasse, Schöckengasse 5, 7253 Renningen 2

Gatch the Spiders DM 10,– in TI-Basic Schein + Leercas. + RP M. Möller, Eschenweg 3, 2150 Buxtehude

---

Mini-Assembler mit viel Software 3 Lit. zu verkaufen: VB 100 DM Axel Ludwegis Tel. 04171-71841

Verkaufe: Mini-Assem. (100 DM) und MASH-Mod. (30 DM) A. Ludewigs, Pestalozzistr. 7, 2090 Winsen/L.

Verkaufe alles im TOP Zustand TI 99/4 nur 200,– Ex. Basic nur 150,– Mini-Mem. nur 200,– Schamodul nur 80,– Personal-Report nur 100,– div. Spielmodule je 50,– Fachbücher für MM und EX-B Alles zusammen nur 600,– Klaus-Dieter Ackermann Lüneburger Str. 11 2125 Salzhausen Telefon am Tage 040-321023166 abends 04172-8621

Suche defekten TI59 incl. Netzteil für Umbau. Paolo Pirillio Battonnstr. 30, 6000 Frankfurt 1

Verk. Ex-Basic orig. TI mit engl. Handbuch 140,– Suche Mini-Mem + Handbuch Tel. 0231/20837

Verkaufe: Modul-Othello 30,–, Minus-Mission 20,–, Alien-Addition 20,– gegen NN Tel.: 02245/3983

Verk. zu TI-99/4A ERW. 32 KR „Atronic" extern, DM 150,– Ext. Basic (o. Handbuch) DM 130,– „A-Mazeing" u. „Hustle" zus. DM 60,– Tel. 040/6523901

Achtung: Suche Terminal Emulator II. Angebote bitte an: M. Lipowski. In der Wanne 165 Gastrop-Rauxel 4

BÖRSE

Top TI 99/4A + P-Box + Mon. + 2 Laufw. Viele Module u. Andw.-Progra. / wie Multi Pl. / TI write etc. H. Fluri, CH 065228995

Suche Programme in X B. und Tib. Ich tausche und kaufe. Liste an J. Soukup am Brahmberg 26, 2358 Kattendorf

---

99/4A+X-Basic (dtsch + engl. / + E/A(neu) torig Thermo-Printer + Basic – X-Basic-Kurs; Finanzberater; TI99-Special 1000 DM VB Tel. 06131/22 25 79

Verkaufe Ext. Handbuch (engl.) für DM 10. Tel. 0203/772112 Jacob Joussen Fasanenstr. 4 4100 Duisburg 1

Suche Monitor und Akkustikkoppler. Biete Schach-Modul und Geld Tel. 0211/377795

TI99:Software/Bücher/Modul 40 Spiele auf Org. cass. 35,– 6 Datenvprg. (Tab; 15,–; Farbbildmusterprg. + Anl. 9,–; Modul „Wumpusjagd" + Anl. 25,–; Bücher üb. Soft-Hardware – 50 % Serviceunterlagen kompl. 15,– Chris Baumgärtner, Pf. 55, 8679 Oberkotzau

---

TI99/4A, X-Basic, Joys. Rek. + Kabel, Literatur + Adventur + Anwprogr. zus. nur 550,– DM, ab 18 h. H.A. Martin, 06898-43646

TI-User im Raum 1A/RA suche Anwenderprogr. Literatur, Tips, alles über Denti. Für ext. Rec. u. Gpiuoa Tel. 07245/3155

–)–

Wegen Systemwechsel alles von meinem 99 günstig, evtl. Liste oder Tel. 09151/95153

Nappsoft Tausch + Programme in TI + Ex. Basic. Liste an Ralf Neumann, Keplerstr. 21, 6680 Neunkirchen 7

Verkaufe 35 Spiele in Basic für 20 DM. 20 DM im Umschlag an: Christian Brock, Massener-Kirchweg 39d 4750 Unna

---

Original TI/Extendet Basic mit Handbuch DM 190,– O. Snishcek, Tel. 069/618262

TI99/4A mit Ext. B./Cass.-Recorder/Kabel/Joysticks und reichhaltiger Software sowie Literatur für 550,– DM zu verkaufen. Tel. 030/391 17 86

Verkaufe Speech-Synthesizer (100 DM) + Alpiner (25 DM) und anderes Frank Henning, Waldstr. 12, 6460 Gelnhausen 2

Suche defekte TI Peripherie. Zahle ja nach Zustand u. Art. Dieter Heike. Tel. 02825/1394 Nur nach 18 Uhr

---

TI99/4A + X-Basic + neuw. Editor-Assembler-Paket + Joystick+ Recorder+Kabel+ Module+viel Software& Literatur. VB 710,– Thomas Schulze (030) 7726346 Morgensternstr. 29, 1 Bln 45

Verkaufe Computerkurs-Heft 1–39 EM 80,– Tel. 0201/ 511225

Entferne Listschutz! Pro Prg. auf Cas. nur 3 DM + 2 DM Porto. R. Bauer, Kelterstr. 16, 6962 Ad.-Sennfeld

---

Suche dringend Drucker + Schnittstelle für TI/99 4A Th. Schmidt, S160DN, Eiswiese 7

Suche Tauschpartner für TI und Ex-Basic Programme. Andreas Scholz, Londoner Ring 6, 6700 Ludwigshafen

---

Ti-Module günstig: z.B. Miner 2049'er DM 40.–, Parsec, Buck-Rogers je DM 25,– (weitere Module a.A.); Dynamics-Prg's für nur DM 6,–; Suche Anfragen an A. Bosbach; Hand-Erhardstr. 56; 8000 München 83 (Rückporto!)

Verkaufe Modul Ext. Basic und Mini Memory, neuwertig, komplett mit Handbüchern zu je 200,– DM, VHB. Alexander Preuß, Sievekingdamm 17a, 2000 Hamburg 26, T. 040/2503478

Suche für TI99/4A gute Softwareprogr. auf Kassette, auch in Extended Basic! Tel. 02331/301640

Wir suchen im Raum Wien TI-Freunde an Erwin Kinslechner, Eibesbrunnerg. 1/10, 1120 Wien, Österreich

---

Oldies B. Goodys DM 8,– Basic f. Anfänger DM 5.– Kasset. Rec. Kab. DM 15,– Div. Bücher, alles original, Tel. 06638/1503

Verk. TI99/4A, Rec. Kabel Mod. Car Wars, Joy + Adap. Lit., Cass. Software (Basic Kurs) VB 280,– DM. Tel. 02948/1541

Dringend zu verkaufen. 2 unbenutzte Cass. Recorderkabel für 15,– je Stück VB Tel. 06055/4684

TI99/4A+EBAS 350 DM Module Stat, RepGen, Dat-Verw je 80,– Minus Mission, Alien Addition je 25,–, Rec-Kabel 15,–, TI-Joyst. 35,– dt. Handbuch ExBas 30 DM, kompl. 645 DM K. Lindner, Weilerstr. 7, 5303 Bornheim 4, Tel. 02227/7040

Verk. TI99/4A+P-Box+ Diskette 32K Erw.+Rs232+ Drucker+Tisch+Journal+ Multiplan+Datenverw.+ Lager+Spiele VHB 2500 DM Grüner, Delpstr. 1, 6940 Weinheim, T. 06201/15413

An alle TI99-Anwender Kassette mit 4 Info-Progr. vollgepackt mit tollen Basic u. Ediass-Progr. DM 5 in Umschl. bei Kauf Vergütung an D. Karbach, Remscheider Str. 18, 5650 Solingen 1

Verkaufe Ti99/4A mit Zubehör. Willi Nussbaum, Mozartstr. 5, 5013 Elsdorf, Tel. 02274/6722

Verk.: Mini-Assembler (Radix) 100 DM; baue Speichererw. 180 DM Suche: Schaltpläne von Konsole und Peripherie; Listing von DSR; HW Untch Tel. 0841/77855

Wer tauscht Scott-Adventure-Kassetten Habe: Count; Voodo und Adventure–Land Anfr. an: Vale 06421-44408

Tausche Return to Pirates Isle f. Monnwepster Modul Tel. 81592, H. Arnfelt, Waldenburgerstr. 8, 2850 B'haven

---

Verk. Softw. Ex + Ti brandneu Galactica. Inex. Kassette 10 DM-Schein an H. Arnfelt Waldenburgerstr. 8, 2850 Bremerhaven

---

Verk. Orig. TI 32Z Thermodrucker inkl. 4 Rollen Papier anschlfertig. VHB DM 450,– Tel. 07631/13732

TI-99/4A Suche Disk mit 32 KB Erweiterung Angebot an: Markus Theimer, Schwedenstr. 2, 6203 Hochheim

---

Verk. TI99/4A + Ext Box + 32kRAM+V24+DSK+Contr +Schrank+XBasic+TI-Writer+ Text-Dateiverwalt+Datenverwalt-Analyse+Versa-Liste+ Number-Magic+Joysticks+ Literatur gg. Gebot: A. Ruff, Haußerstr. 140, 7400 Tübingen

---

12345678901234567891234 Verkaufe TI99/4A+P-Box+ Floppy+Controller+32K+ Speech-Synthy+X-Basic+ E/A (dt. Handb.)+versch. Progra.u. Literatur VB 2200 DM, auch einzeln Martin Ottomeier, Werthfeldstr. 3, 42 Oberhausen 12, Tel. (0208) 893532

Suche Kontakt zu Besitzern von APESOFT-GRAFIC-BASIC zwecks Informationsaustausch! Verkaufe: TI-Basic/Ext. Basic für Anfänger für 15 DM! MATTHIAS ORF, BIRKENALLEE 34, D-3507 BAUNATAL 1, TELEFON: (0561) 497990

Verk. TI-Module: Parsec, Statistik, Datenverw. u. Analyse je 30,– DM Tel. (0821) 419121

Verkaufe Basic-Kurs, Finanzberater und Dynamics-Spiel Darts zu je 10,– Tel. 08851/5528

Extended Basic + Ex.Handbuch+50 Programme = 200,– DM, 2 Joysticks (original) = DM 49,– Tel. (09281) 95166

Verk.: TI99/4AText. – Bas. + Joyst., Buchungsjournal, Datenverw., Statistik, Schachmeister+ 6 Spielmodule + 6 Cassetten + div. Lektüre Tel. 06893/1553 J. Peter, Ponsheimer Str. 4, 6601 Eschringen

---

Suche: Modul TEXT & DATEIVERWALTUNG! Dringend! Angebote an Matthias Orf, Birkenallee 34, D-3507 Baunatal 1 Tel. 0561/497990

BÖRSE

TI-Club Münster e.V. bietet
eigene TE-2 Übersetzung fü
für nur 5 DM. Tel. 0251/
45845! Selbstkostenpreis!

Verk. original TI-Laufwerk
für 200 DM. Tel. 0251/
664684

Verk. TI99/4A i. Orig. Verp.
+ Ex.B. + RS232-2P+Joyst.
+ Rec.+Druck. EP44 + Lit.
1200 DM Raum Saar
06834-42409

Verkaufe Mult. Pl. an 200
Orthello Tomb. je 20 Musik
Schach je 50 Editor 100
Adv + 12 Games 150 Buchg.
100 Stat + Text + Pers
je 50 + div.
Tel. 0212/56537

Verkaufe Parsec Hunt The
Wumpus TI Invaders Car
Wars für je 30,—
Tel. 0212/56537

TI-RECHNER-Menuegesteu-
ertes Programm mit 14 Re-
chenarten incl. Cass. +
Porto nur DM 10,—
öS 80,— in Umschlag an:
D. Karbach, Remscheider
Str. 18, 5650 Solingen

TI99-Box, 32K, RS232, 2
Diskst. Ass., Exb., Min. Text.
u. Datenver., Dat. u. Analyse,
Diskmanager, Spr.-Syn.
Tel. 030-7724897
VB 2600,— DM

Zu Verkaufen: TI99/4A
1 Modul Verwaltung
Text + Datei Mini Mem.
RS 232 Extended Rueegg
Bern
Tel. 031 559292/243232

TI-Box mit Laufwerk +
Controler + RS232 für
1000 DM TI-99/4A + Ex-
Basic + 5 Spielm. + 2
Anwendungsm. + Rekorder
650,— Super Sketch,
Grafiktablett für 150 DM
069/685142

Suche Kontakt zu anderen
E/A Usern und tauscbe E/A
Prgs Ingo Kuning,
Obernhainer Weg 1,
6380 Bad Homburg,
Tel. 06172/33879

Verk. MBI Drucker-Inter-
face (Centronic) für nur
100 DM. Funktioniert tadel-
los. Ingo Kuning, Obern-
hainer Weg 1,
6380 Bad Homburg
Tel.: 06172/33879

Verkaufe billig TI/99 4A
+ Extended Basic + Kassetten-
recorder + Spielmodule
und Buchmaterial + Joy-
sticks
Jörg Schlatter,
Tel. 02271/64514

Wegen Systemwechsel
TI99/4A+XBas+Peri-Box
(CPS99/Atronic(1LW-DS/DD
32K, Centr. 2xRS232)
4Mon.Gar.+ 9 Bücher+Ti-
Revue(compl)+Software+
Rec.mit Kabel+ 1 Joystick
komplett DM 2.200,— VHB
Tel. 04191/60215 ab 19.30

TI EXTENDED BASIC (dt.
Nachb.)+ 99Special II+TI
Intern+CHIP Listings +Pro-
grammcass.+Zeichenschriften.
PREIS: 260 DM
Holger Wenzel
Tel. 06138/8395

TI- und Ex-Basic-Programme
Info g. Rückporto. B. Knedel,
Tulpengasse 16, 3171 Wey-
hausen. Tel. 05361/71187

Suche gute E/A Pgms. im
Tausch. Schickt Eure Listen
zu: M. Zieske, Griesingerstr.
152, 1000 Berlin 20

Achtung: Suche gebrauchtes
Grafik-Tablet evtl. mit Druk-
kerprogramm.
Tel. 08453/2585 ab 17 Uhr

Ich verkaufe meinen TI99/4A
(kompl.) für 2000,— DM
inklusive Literatur
T. 0201/402 395, nach
17 Uhr

Verkaufe TI99/4A + Zubehör
f. 200 DM
Tel. 05223-86622

Verkaufe TI-99/4A+Recorder
Kabel + 6 Bücher (Basichand-
buch, Programmbuch, Tips
und Tricks, ...) + Software für
250,— DM
Telefon: 0761/39525

Suche externe 32K-er W.,
evtl. mit Centronics-Schnitt-
stelle
Burkart, Tel. 06721/41859

Suche Sprachsynth. (bis 100)
Schachmod.
Martin Grünwald
Brückenstraße 4
8034 Germering

Org. ExBox m. Disclaufw.
Controller u. Manager 2
nur DM 749 Mod. Daten-
verw. DM 40. J. Weiß,
3380 Goslar, Franckestr. 14

TI99/4A Grafiktablett
Supersketch (DM 140,—)
Tel. 02323-51417

Suche Leute, die Lust haben
mit mir ihre Programme zu
tauschen. Würde mich riesig
freuen, wenn sich möglichst
viele melden. Liste an Carlos
Jarques, Ludwigsburger-
str. 14, 7257 Ditzingen 1
Tel. 07156/3491. Ich be-
sitze ca. 200—250 Spiele
C. Jarque, Ludwigsburger-
str. 14, 7257 Ditzingen 1

Verk. TI99/4A, org. ext.-
Basic + detu. und engl.
Handbuch und 2 Bücher,
Schachmdl., Recorder,
sämtl. Kabel und klein TV,
zus. 550,— DM. Melden bei
A. Ebel, Westhoffstr. 15,
4630 Bochum oder Sa + So
bei Diederich 02302/71299

Verk. f. TI; Othello, Mind
Chall., Tombst. City, Chrish.
Trail f. je 25.—
Tel. ab 18 h
0821/3156399 Dieter

Verk. Top TI 99/4A +
Cass. Aec. + Rec. Kabel +
GP100A + Kabel + P.-Box
Kompl. mit int. Contr.,
Disk-, 32K, RS232+ TI-Joyst.
+ Ex-Basic + Ed. Assem. +
Sprachsynt. + 8 Mod. + 10
Disk. + viel Literatur
Preis = VHS M. Sandmann,
Im Braukes, 4630 Bochum
Tel. 0234/382596

TI99/4A=Club der Toten=
VC-20 Aktivitäten: Club-
magazin, Clubtreffen, Club-
info, Softwarebibliothek,
Einkaufsgemeinschaft.
Jahresbeitrag 12 DM!!
Mehr Aktivitäten auf
Wunsch!
Info gegen Rückporto.
Adresse: CDT, Am Hang 5
D-2054, Geesthacht =
Club der Toten

TI99/4A-Drucke Progr./
Dateien — egal wie lang
(DM 10,—)
Schreibe Listings auf Disk/
CAS.
Biete Progr. versch. Bereiche
aus Zeitschriften (z.T. Masch.
Progr.) Verkaufe div. Module
Adress: B. Dusny, 5205
St. Augustin 1, Mülldorfer
Str. 31, Tel. 02241/330700

Verkaufe TI99/4A orig. verp.
Cass. Rek. Kabel, Modul,
Basic-Lehrg. Mod. Daten-
verw. u. Analyse, NP 700 DM
VK 500 DM
W. Kraatz, Heldstr. 3,
5489 Barweiler, Tel. Mo+Mi
ab 20 h. Tel. 02691/7836

!!! Suche !!!
xxxxxxxxxxxxxxxxxxxxxxx
Suche Spielmodul im Tausch
gegen TI-INVADERS
0221/697254

***QUBIE & THE CUBES**
*Superspiel f. TI 99/4A (Ex *
*basic, ähnl. Q-Bert, 3D- *
*Graphik, DM 25,— incl. *
Prgmcassette, Porto u. *
*Versand Softwareinfo gg. *
*DM 1,— *
*P. Rieger, Talstraße 64 *
*6750 Kaiserslautern ******

Suche TI-Besitzer im Raum
Leonberg und Umgebung.
Ich weiß, daß es welche
gibt!! Meldet Euch bitte
bei Carlos Jarque,
Ludwigsburgerstr. 14,
7257 Ditzingen 1
Tel. 07156/3494

ACHTUNG ASSEMBLER-
FREAKS!
Suche Kontakt zwecks Aus-
tausch von Informationen
und Assemblersoftware.
Weitere Interessen:
DFÜ und Hardwareeigen-
bau
G. Morgenthaler, Angelstr.
7143 Vaihingen/Enz

Verk. TI 99/4A + X-Basic +
div. Module + Software +
Recorder + Kabel + Joy-
sticks + Literatur
VB 550 DM, Floppy + Con-
troler 650 DM
C. Bock, Oberstraße 106,
3202 Bad Salzdetfurth
Tel. 05063/551

ALPHA — DATEK V 1.0
99'er Spitzen Software
Ihre profi Datenbank f. Ihren
TI 99/4A. Ein Super Prog.
mit 40 Zeichen pro Zeile,
SUPER schnellen Sortieren
und echte Kleinb. durch E/A
Unterprof. und vieles mehr f.
wenig Geld. Info by U.
Brüsseler ab 20 Uhr,
4050 Mönchengladbach 3,
Wetschewell 125 d,
T. 02166/53457

Viersener TI-User bitte bei
mir melden.
Peter Voigt
Breslauer Str. 20
Viersen 11
Tel. 56520

ACHTUNG BASTLER!!
TI 99/4A (leicht defekt)
& PAL & Netz nur 100,—
Recorderkabel 15 DM, Hand-
bücher (EX+TI) 10,— &
5.— DM, vollbespielte
Programmkassetten nur
8,— DM
Dirk biehl, Hyazinthenweg
15, 6680 Neunkirchen 7

BÖRSE

Zu verkaufen: TI Expansion Box, V24-Karte, DSK-Drive (int.), DSK-Contr.-Karte und DSK Manag. Mod. gegen Gebot bei: Albrecht Eger, Aeulestr. 15, 7400 Tübingen

---

Die BESTEN TI99/4A Programme in Extened Basic: James Bond 007 – Im Angesicht des Todes nur 10,– DM + Leerkassette an Pat. Schmitz, Am Zehnthof – 4, 5480 Remagen – Unkelbach ...
Weitere Top-Programme: Liste gegen frank. Rückumschlag!!

TI 99/4A: EINMALIGES ANGEBOT!!! TI+Zeitschriften + 6 TI-Revues ORIGINAL VERPACKT!! nur 150,– DM. MM + Handbuch + Buch „Einführung ins MM" + 40 Prg. auf Kassette + Anleitungen nur 200,– DM.
Rekorder + Kabel a.A.
Superbücher usw. a.A.
Meldet Euch bei: M. Rattelmüller, August-Bebel-Str. 11, 6454 Bruchköbel 1
Tel. 06181/76654

Verkaufe Minimemory mit dt. Handbuch (200), Orig.-Adventure-Cassetten: Pirate Adv., Savage Isl., Pyramids of Doom (zusammen 50, einzeln 20)
Tel. (ab 19.00): 0221/481053

Suche: billige Spielmodule, anschlußfertige Floppy für TI u. Spiele.
T. Pfister, Talstr. 49, 8770 Lohr

Suche und tausche Anwenderprogramme in X-Basic oder Assembler (Mathe, Physik, Textverarbeitung ...) Alex. Rupp, Kalmanstr. 45, 6600 Saarbrücken

---

Suche das Spiel Unternehmen Galactica für den TI99/4A Angeobte an: Stefan Jennen, 02832 70242

Verkaufe TI99/4A + Recorder + Recorderkabel + viele Programme + viele Computerhefte 250 DM
Tel. 069/614907

Verkaufe komp. Anlage Neupreis 3000,– für 1500,– (Box; Ex-B; Grafiktablett...) T. Nebgen, 069/685142

Popeye auf Cassette! (in extended-Basic!) 10 DM-Schein an S. Altstadt, Hochstr. 23, 6913 Mühlhausen 2

Verkaufe Speichererweiterung 32K intern für Peri-Box 200,– DM.
Tel. 08192/231

Verkaufe TI 99/4A Komplettsystem mit allem, was des Computerianers Herz höher schlagen läßt – auch einzeln. Ruft an! Liste kommt kostenlos. Tel. 08192/231

Ti99/4A, extended B., 3 Bücher, 2 Modul: Video-Games I, Software, Kassettenrecorder + Kabel, 3 Joysticks.
Norbert Wambach
Kais.-Heinr.-Str. 52
5401 Urmitz/RH

Suche preisw. Erw. 32KB Disk usw. Außerdem bin ich an Kontakten zu Usern od. Clubs interessiert (Neuling) K-H Morawe, Karlstr. 4, 4300 Essen 12

Verkaufe meine 10 Original Apesoft-Disketten mit gesamter Software aus 99-Special-Bücher I + 2 + Cerberus: 190 DM;
TI-Module: Demon-Attack, Carwars, Attack, Othello je 25 DM – alle 4 nur 90 DM.
Tel. 08141-23140 – nur von 19-21 Uhr

Verkaufe Drucker GP-100A + MBI-Interface, kompl. für DM 280,–
Tel. 06172/32849

TI-99/4A; P-Box, 32K, Disk, P-Code; RS-232; Ex-Bas.; TI-Writer; Ass. usw. nur: 2500,–
Berdon, Heusenstamm,
Tel. 06104/62185

Suche: Erw.-Box + 32 K + Disk-Laufwerk + Controller für TI 99/4A. J. Kliem, E.-Brandström-Str. 37, 5042 Erfstadt

---

TI99/4A, X-Basic+Handb., P-Box 2 x RS232 + Centronics, 32 KB, Contr. + 1 Laufw., Druckerkabel, TI-Writer, Statistik, ID-Data, Ti-Special I + II
VB, Tel. 06722-71318

Hab ne Menge zu verk. PBox + Dis + 32K + Multip + Soft auch einzeln:
V. Wichert, Linzgaustr. 7, 798 Ravensburg

Verkaufe: RS232-Karte (Orig. TI) DM 250 VB Konsole DM 180, Ext. Basic DM 150; 3fach Modul-Exp. 02041/22661

Verkaufe: Ti99/Ext.Bas+ Recorder+Akustikk.+ TEII. + Hacker-Handb. + Bas.-Kurs für 1200 DM
Tel. 06371/50564

Verk. TI-99/4A + Ex.-Basic + Sreech-Synthesizer + Datasette + ca. 100 Programme + 6 Module
VB 650,–
Tel. 06103/81815

99/4A + Basic + P.Box + RS232 + Disk. + EIA + Dataphon u. versch. Module für 1500 DM
Tel. 02171/45724

Verk. Peribox f. TI99/4A m. Flexkabelkarte RS 232 Schnittst.-Karte, Disklaufwerk, 32 KB-Sp. Erw.-Karte + Lit. + Prg. VB DM 950,–
Erwin Wachter,
Gerstbuschstr. 9
6670 St. Ingbert
Tel. 06894/6890

Suche folgende Bücher:
9900 Family Systems Design and Data Book, TMS 9900 Family Software Development Handbuch, 16-Bit Mikroprozessor, Applikationsbuch Bd. 2
Angebote an:
M. Humbert, Saphirring 30, 3008 Garbsen 1
Tel. 05131/93511

TI99/4A + Ext. Bas. + Ext. Bas Plus II + Cass.-Rec + Kabel + 3 Bücher + 3 Mod. + ca. 16 Kassetten mit Prog. VHB: 900,–
Tel. 06150/3060

!!! ADVENTURE !!!
Suche billigst original TI-Adventure (oder Infocom) Angebote unter: 0221/697254

0000000000000000000000
Suche Touch Typing Tutor . (Schreibmaschinenlehrgang) 0221/697254

Sensation: Rex-soft Katalog nur 1,– DM!!!
50 % des NP 000000000000
Info unter: 0221/697254

TI 99/4A + Box mit 32K + Laufwerk, X-Basic, ED/Ass, Minimem, Schach, Statistik + div. Lit. VB 1200,–
Otmar Scheitl, Hans-Sachs-Str. 76, 8706 Höchberg,
Tel. 0931/40486 ab 17 h

Zu verk. Neue MBI Interface-Centron. DMK 150,– & Module Buchungsjourn. DM 70,–
Tel. 069/2566225

T199-Zubehör: Ext.-Basic, Periph.-box, RS232 Schnittstelle, Laufwerk + Controll. G. Renker, Herstadtring 151, 4630 Bochum 1

Komplett TI99/4A Konsole + PBox + RS232 + Ext. Basic + Laufwerk + 32 Kram + Minimemory + Schach + Lit.
Tel. 07231/41436

Verk. TI 99/4A + Ext.-Basic + Rek.-Kabel + Joyst., Buchungsjournal, Datenverw. + Anal., Statistik, Schach usw., div. Lit.
Peter Jürgen, T. 06893/1553

Verkaufe TI99/4A + X-Basic IIPlus mit deutschem Handbuch + Rec.-Kabel + Parsec + Joystick für 500,–
Tel. 06103/25571

Verkaufe: RS232 ext. Schnittstelle (150) TI-Basic & Extended Basic Buch (20)
07552/1880 od. 5420

TI99/4A 10 Ex-Bac. Prg. auf Cass. nur 15 DM
Adrs.: Metin Gurer, Bahnhofstr. 70, 7123 Sachsenheim 1

Suche externe RS232 Schnittstelle mit durchgeführtem Bus
Tel. 07821/7570

TI 99/4A + XBasic + Minimen + Schachm. + Adventurem. + Software + Recorder + Joyst. + Literatur. VB 850,– DM
Tel. 0551/37 20 28

TI 99/4A, viel Literatur (neu über 200,–) + Programme zus. 150,– + orig. TI-Joyst. 30,– + ext. RS232 (2 Ports) 250,– + brother-Drucker EP22 170,– + X-Basic (dt. Handb.) 170,– + Dateiverw.-Modul 40,–; kompl. 680,– alle Preise VB
Tel. 0421-534176

M.P.7!! Ein Strategiespiel für Gehirnakrobaten und solche, die es werden wollen. Ein „Muß"!
Info: 0221/697254

Suche TI-User in Köln zwecks Erfahrungs- und Softwareaustausch. Meine Konfiguration: Console, Kassettenrekorder, XB. Tel.: 0221/697254

Darst. Ein Superspiel von Synamcis zum Super-Preis von: 5,– DM !!! (Natürlich orig. Info: 0221/697254

HeGa-Soft

Software, Hardware und Literatur
Postfach 1201   8044 Lohhof

>>>>>>>>>>>>>>>>>>>>!!! Es gibt uns noch !!!<<<<<<<<<<<<<<<<<<<<
Rechtzeitig vor Weihnachten das aktuelle Angebot für den TI 99-4 A
>>>>>>>>>>>>>>>>>>>>############################<<<<<<<<<<<<<<<<<<<<

1.) Module (naturgemäß nur begrenzte Vorräte)	
TI Schach	DM 60,--
Othello	DM 40,--
Datenverwaltung und Analyse	DM 60,--
Statistik	DM 40,--
Buchungsjournal	DM 60,--
Microsurgeon	DM 65,--
Moonsweeper	DM 65,--
Alien Addition	DM 15,--
Demolition Division	DM 15,--
Meteor Multiplication	DM 15,--
Minus Mission	DM 15,--
Dragon Mix	DM 15,--
Alligator Mix	DM 15,--

2.) Cassettensoftware	
Personal Financial Aids	DM 15,--
Mini Writer (für Mini Memory)	DM 128,--

3.) Zubehör	
Rekorderkabel	DM 10,--
Joystickverlängerung ca.3m	DM 10,--
Joystickadapter	DM 25,--
Druckeranschlußkabel (PIO)	DM 79,--
Disketten SS SD, 10 Stück Packung incl.Labels	DM 35,--

4.) Hardware	
Drucker MT 80 bzw BX 80	DM 798,--
Disclaufwerk 6106	DM 295,--
Multiboard TI auf RS 232	DM 150,--

5.) Diskettensoftware	
iD Data	DM 119,--
iD Konto	DM 119,--
iD Profi Set (iD Data + iD Konto)	DM 198,--
Graphic Master	DM 99,--
3 D World	DM 99,--
TI Forth	DM 148,--
Ex-Basic Loader für Forth	DM 25,--
Diskkatalog mit Autostart	DM 25,--

6.) Bücher	
99 Spezial 1	DM 48,--
TI Basic/Extended Basic deutsches Handbuch	DM 48,--
Deutsches Handbuch zum Editor/Assembler Modul	DM 98,--
deutsche Kurzanleitung zum TI LOGO (28 Seiten)	DM 19,80
ausführliches deutsches Handbuch zum TI LOGO(218 Seiten)	DM 59,--

Lieferung solange Vorrat reicht. Bestellungen bitte nur schriftlich. Lieferung per Nachnahme. Versandkostenpauschale Inland DM 5,--. Ab DM 100,-- Bestellwert kostenfrei. Ausland generell tatsächliche Postgebühren. Druckfehler bzw. Irrtum vorbehalten.